



UNIVERSIDADE FEDERAL DO PARÁ
NÚCLEO DE INOVAÇÃO E TECNOLOGIAS APLICADAS A ENSINO E EXTENSÃO
PROGRAMA DE PÓS-GRADUAÇÃO CRIATIVIDADE E INOVAÇÃO
EM METODOLOGIAS DE ENSINO SUPERIOR
MESTRADO PROFISSIONAL EM ENSINO

ÁRLLON CHAVES LIMA

METODOLOGIA 70S

**UMA PROPOSTA DE ENSINO E APRENDIZAGEM
PARA DISCIPLINAS INTRODUTÓRIAS
À PROGRAMAÇÃO**

BELÉM - PARÁ
2020

Árllon Chaves Lima

METODOLOGIA 7Cs
UMA PROPOSTA DE ENSINO E APRENDIZAGEM PARA
DISCIPLINAS INTRODUTÓRIAS À PROGRAMAÇÃO

Dissertação apresentada ao Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior da Universidade Federal do Pará, como requisito parcial à obtenção do título de Mestre em Ensino. Área de Concentração: Metodologias de Ensino-Aprendizagem. Linha de Pesquisa: Inovações Metodológicas no Ensino Superior (INOVAMES).

Orientador: Prof. Dr. Marcos Monteiro Diniz
Coorientadora: Profa. Dra. Marianne Kogut
Eliasquevici

BELÉM-PARÁ
2020

O46a Lima, Árlon Chaves

Metodologia 7Cs: uma proposta de ensino e aprendizagem para disciplinas introdutórias à programação/Árlon Chaves Lima. Belém: UFPA, 2020.

169f.

Dissertação (Mestrado Profissional em Ensino) – Universidade Federal do Pará. Belém, 2020.

1. Metodologia 7Cs 2. Algoritmos 3. Ensino Aprendizagem 4. Aprendizagem Significativa.

I. Título.

C.D.D. 371.102

Árllon Chaves Lima

METODOLOGIA 7Cs
UMA PROPOSTA DE ENSINO E APRENDIZAGEM PARA
DISCIPLINAS INTRODUTÓRIAS À PROGRAMAÇÃO

Dissertação apresentada à Universidade Federal do Pará, como parte das exigências do Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior, Mestrado Profissional em Ensino, para a Defesa de Dissertação.

Orientador(a): Prof. Dr. Marcos Monteiro Diniz
Coorientadora: Profa. Dra. Marianne Kogut Eliasquevici

RESULTADO: (**X**) APROVADO () REPROVADO

Data: 08/05/2020

Prof. Dr. Marcos Monteiro Diniz (Orientador)
Programa de Pós-Graduação em Criatividade e Inovação em
Metodologias do Ensino Superior (PPGCIMES)

Prof^a. Dr^a. Marianne Kogut Eliasquevici (Coorientadora)
Programa de Pós-Graduação em Criatividade e Inovação em
Metodologias do Ensino Superior (PPGCIMES)

Prof.^a. Dr^a. France Fraiha Martins
Programa de Pós-graduação em Docência em Educação em
Ciências e Matemáticas (PPDEGM)

Prof.^a. Dr^a Silvana Rossy de Brito
Universidade Federal Rural da Amazônia (UFRA)

BELÉM-PARÁ
2020

Dedico a minha família, mãe, pai e irmão, que tanto lutam e trabalham para que eu possa conquistar meus objetivos e sonhos.

AGRADECIMENTOS

A Deus por realizar esse grande sonho em minha vida e por permitir que pudesse encontrar no caminho pessoas tão incríveis, como meus colegas e professores do mestrado.

À minha família que sempre estão na torcida pelas minhas conquistas e em especial a minha mãe **Terezinha Chaves de Lima**, meu pai **Antônio Cavalcante de Lima** e ao irmão e parceiro **Arllan Chaves Lima**, que trabalham incansavelmente para sustentar meus maiores sonhos.

Aos meus orientadores, **Prof. Dr. Marcos Monteiro Diniz**, um excelente profissional e um ser humano incrível, e **Profa. Dra. Marianne Kogut Eliasquevici**, por toda ajuda e pela enorme parcela para qualidade do trabalho desenvolvido. Obrigado por terem acreditado no meu potencial e principalmente, pelos incríveis ensinamentos e conhecimentos compartilhados. Sou muito grato a Deus por ter feito parte dessa empreitada com vocês.

Aos amigos do mestrado, por todos os dias que lutamos e juntos conseguimos vencer! Levo dessa linda experiência pessoas incríveis, em especial, representando a turma, as amigas Rosa Ametista e Andreza Jackson que estiveram ao meu lado em muitos momentos difíceis, obrigado por tudo.

Ao PPGCIMES, pelas grandes experiências e vivências que, com certeza, mudaram quem eu sou, tornando-me uma pessoa e profissional melhor. Sou eternamente grato pela oportunidade que me foi dada.

“Seja forte e corajoso! Não se apavore nem desanime, pois o Senhor, o seu Deus, estará com você por onde você andar”.

Josué 1:9

RESUMO

O presente trabalho de dissertação apresenta a concepção de uma metodologia de ensino e aprendizagem para disciplinas introdutórias à programação, intitulada Metodologia 7Cs, composta por sete dimensões (Compreender, Conceber, Completar, Compatibilizar, Corrigir, Construir e Criar) e que tem por objetivo facilitar o processo de aprendizagem dos conteúdos básicos de algoritmos, de modo a minimizar dificuldades dos discentes de graduação de cursos que contemplem disciplinas introdutórias à programação, disciplinas essas frequentemente responsáveis por altos índices de retenção e evasão nos cursos de graduação na área de TI. A concepção de tal metodologia se baseou em princípios da Teoria de Aprendizagem Significativa de Ausubel (2000) e teve inspiração inicial no trabalho de Campos (2010). Foram realizadas observações participantes em dois diferentes cursos de graduação da Universidade Federal do Pará (UFPA) e um teste piloto no Programa de Nivelamento em Algoritmos (PNA) da mesma universidade. Em seguida, a partir das observações e dados obtidos, a metodologia foi aplicada na disciplina Algoritmos, do Curso de Bacharelado em Ciência da Computação, também na UFPA. Verificou-se a possibilidade de utilização da metodologia de forma flexível, com exequibilidade, tanto em cursos de nivelamento, quanto na disciplina de algoritmos em nível de graduação. Elementos observados indicaram potencial da Metodologia 7Cs para auxiliar na redução das dificuldades na compreensão dos conceitos básicos e no desenvolvimento de habilidades associadas a disciplinas de algoritmos, com a verificação, ao longo das aplicações, de envolvimento mais ativo dos estudantes e desenvolvimento de habilidades específicas relacionadas à construção de algoritmos. Finalmente, foi elaborado um Material de Orientações, com descrição sucinta da Metodologia 7Cs e exemplos de possíveis formas de sua aplicação, para ilustrar seus princípios e auxiliar docentes interessados em utilizá-la.

Palavras-chave: Metodologia 7Cs. Algoritmos. Ensino e aprendizagem. Aprendizagem Significativa.

ABSTRACT

This thesis presents the conception of a teaching and learning methodology for introductory subjects to programming, entitled 7Cs Methodology, composed of seven dimensions (Understanding, Conceiving, Completing, Compatibilizing, Correcting, Building and Creating) and which aims to facilitate the learning process of the basic contents of algorithms, in order to minimize difficulties of undergraduate students of courses that include introductory subjects to programming, subjects that are often held responsible for high retention and dropout rates in undergraduate courses in the area of Technology and Information. The conception of such a methodology was based on the principles of Ausubel's Meaningful Learning Theory (2000) and was initially inspired by Campos' work (2010). Participant observations were held in two different undergraduate courses at the Federal University of Pará (UFPA) and a pilot test in the Algorithm Leveling Program (PNA - *Portuguese acronym*) at the same university. Then, from the observations and data obtained, the methodology was applied in the subject Algorithms, from the Bachelor's Degree in Computer Science, also at UFPA. It was verified the possibility of using the methodology in a flexible way, with feasibility, both in leveling courses and in the subject of Algorithms at the undergraduate level. Observed elements indicated the potential of the 7Cs Methodology to assist in reducing difficulties in understanding the basic concepts and in the development of skills associated with algorithm subjects, with the verification, throughout the applications, of more active involvement of students and the development of specific skills related to the construction of algorithms. Finally, a Guidance Material was prepared, with a brief description of the 7Cs Methodology and examples of possible ways of its application, to illustrate its principles and assist teachers interested in using it.

Keywords: 7Cs methodology. Algorithms. Teaching and learning. Meaningful Learning.

LISTA DE FIGURAS

Figura 1 - Procedimentos metodológicos para concepção da proposta de ensino a.....	29
Figura 2 – Fases da Metodologia 7Cs	72
Figura 3 - Tabuleiro do jogo M7C	73
Figura 4 - Cartas Desafio do jogo M7C.....	74
Figura 5 - Carta do jogo M7C projetada	90
Figura 6 - Resolução dos desafios de maneira escrita.....	91
Figura 7 - Grupo discutindo sobre a solução do desafio (dimensão Compatibilizar) ..	92
Figura 8 - Análise das respostas do desafio realizada pelos monitores (dimensão Compatibilizar)	92
Figura 9 - Monitores realizando feedback sobre as soluções dadas por dois grupos (desafio da dimensão Compatibilizar)	93
Figura 10 - Interação entre os alunos para a solução do desafio (dimensão Compatibilizar)	94
Figura 11 - Construção dos primeiros algoritmos dos alunos no curso.....	96
Figura 12 - Disponibilização do código de acesso do jogo.....	97
Figura 13 - Jogo do Kahoot! projetado para os alunos.....	97
Figura 14 - Cores correspondentes aos alternativas da questões	98
Figura 15 - Cores correspondentes aos alternativas da questões	99
Figura 16 - Quantidade de respostas por alternativa.....	99
Figura 17 - Verificação da monitora sobre a quantidade de acertos e erros.....	100
Figura 18 - Questão utilizada para a dimensão Compatibilizar.....	101
Figura 19 - Escolha da ordem das cores para alcançar a resposta correta	101
Figura 20 - Questão utilizada para a dimensão Corrigir.....	102
Figura 21 - Atividade fora de sala de aula da dimensão Colaborar	103
Figura 22 - Execução do algoritmo realizado pelos grupos na atividade	104
Figura 23 - Pesquisa da percepção dos alunos sobre a potencialidade da Metodologia 7Cs	105
Figura 24 - Resultado sobre as habilidades e competências desenvolvidas na metodologia, na percepção dos alunos.....	106
Figura 25 - Resolução da atividade de maneira escrita	108
Figura 26 - Teste das respostas escritas nos computadores	109
Figura 27 - Auxílio do monitor nas atividades em sala.....	110
Figura 28 - Interação entre os alunos durante resolução da lista de exercício	110
Figura 29 - Lista de exercício disponíveis aos alunos.....	112
Figura 30 - Monitor e professora da disciplina auxiliando os alunos.....	113
Figura 31 - Resolução das questões da lista no papel	114
Figura 32 - Alunos discutindo sobre a solução das questões	115
Figura 33 - Quadro magnético com as habilidades e competências da fase Descoberta	116
Figura 34 - Post-its usados para destaque das habilidades e competências	116
Figura 35 - Post-its com comentário de um dos alunos.....	117
Figura 36 - Resolução da lista de exercício de maneira escrita	118
Figura 37 - Execução das soluções no computador.....	119

Figura 38 - Resolução da lista da dimensão Construir	121
Figura 39 - Formação dos grupos para o desenvolvimento da atividade	122
Figura 40 - Grupo construindo um comando de questão no papel	123
Figura 41 - Interação entre os integrantes do grupo para a solução da atividade	124
Figura 42 - Monitor supervisionado e auxiliando os alunos	124
Figura 43 - Capa do Material e orientações	127
Figura 44 - Dialogando com os professores	132
Figura 45 - Códigos de algoritmos	133
Figura 46 - Ilustração das fases da Metodologia 7Cs	134
Figura 47 - Sumário do material de orientações	135

LISTA DE QUADROS

Quadro 1 - Criatividade do produto	27
Quadro 2 - Síntese das principais dificuldades no ensino-aprendizagem discutidas por alunos de programação, mencionadas na literatura (Continua...).....	33
Quadro 3 - Síntese das dificuldades inerentes aos professores mencionadas na literatura	35
Quadro 4 - Requisitos para pesquisa de metodologias	47
Quadro 5 - Propostas metodológicas para o ensino de algoritmos (Continua...).....	49
Quadro 6 - Pontos levantados a partir das contribuições da literatura	51
Quadro 7 - Requisitos para concepção da Metodologia 7Cs	64
Quadro 8 - Descrição da Metodologia 7Cs (continua...)	67
Quadro 9 - Categorias de aprendizagem.....	69
Quadro 10 - Fases da Metodologia 7Cs	70
Quadro 11 - Exemplo de exercício da dimensão Compreender	76
Quadro 12 - Exemplo de exercício da etapa Conceber	77
Quadro 13 - Exemplo de exercício da dimensão Completar	79
Quadro 14 - Exemplo de exercício da dimensão Compatibilizar	81
Quadro 15- Exemplo de exercício da dimensão Corrigir	83
Quadro 16 - Exemplo de exercício da dimensão Construir	85
Quadro 17 – Etapas para dinâmica da dimensão Criar	122
Quadro 18 -- Requisitos funcionais do material de orientações	129
Quadro 19 - Requisitos estruturais do material de orientações	130
Quadro 20 - Descrições das seções que compõe o material de orientações.....	131
Quadro 21- Ferramentas presentes no Grupo e suas funcionalidades.....	137

LISTA DE GRÁFICOS

Gráfico 1 - Escolha pelos alunos das habilidades e competências da fase Descoberta	117
Gráfico 2 - Escolha pelos alunos das habilidades e competências da fase Aperfeiçoamento.....	120
Gráfico 3 - Escolha pelos alunos das habilidades e competências da fase Consolidação	125

SUMÁRIO

INTRODUÇÃO	15
JUSTIFICATIVA.....	18
Primeiras inquietações e motivação	18
Transformação das inquietações em busca de mudanças.....	20
Processo e escolha do produto	21
Inovador e Criativo.....	24
FLUXO METODOLÓGICO DA PESQUISA.....	28
ESTRUTURA DA DISSERTAÇÃO	30
1 ENSINO E APRENDIZAGEM DE ALGORITMOS E SEUS DESAFIOS	32
1.1 DIFICULDADES NA DISCIPLINA DE ALGORITMOS NO ENSINO SUPERIOR.....	32
1.2 A IMPORTÂNCIA DA PRÁTICA PARA A DISCIPLINA ALGORITMOS.....	35
2 APRENDER DE FORMA SIGNIFICATIVA	38
2.1 APRENDIZAGEM SIGNIFICATIVA	38
2.2 APRENDIZAGEM SIGNIFICATIVA E O ENSINO DE ALGORITMOS	43
3 CONCEPÇÃO DA METODOLOGIA 7CS.....	46
3.1 METODOLOGIAS DESENVOLVIDAS PARA O ENSINO E APRENDIZAGEM DE ALGORITMOS.....	46
3.2 TRABALHO INSPIRADOR	52
3.3 OBSERVAÇÃO PARTICIPANTE	52
3.3.1 Relato da primeira observação participante.....	54
3.3.2 Relato da segunda observação.....	57
3.4 RELATO DE PESQUISA EXPLORATÓRIA	60
3.5 REQUISITOS PARA CONCEPÇÃO DA METODOLOGIA 7CS	63
4 CARACTERÍSTICAS DA METODOLOGIA 7Cs	65
4.1 DESCRIÇÃO DA METODOLOGIA 7CS	65
4.2 EXEMPLO DE APLICAÇÃO DA METODOLOGIA 7Cs	72
5 APLICAÇÃO DA METODOLOGIA 7Cs.....	89
5.1 TESTE PILOTO DA METODOLOGIA 7Cs.....	89
5.1.1 Apresentação da Metodologia 7Cs: teste do Jogo M7C.....	90
5.1.2 Desenvolvendo as fases.....	94
5.2 APLICAÇÃO DA METODOLOGIA 7Cs EM DISCIPLINA DA GRADUAÇÃO.....	107
5.2.1 Um teste da forma de aplicação: primeiras impressões.....	108
5.2.2 Novo teste	111
5.2.3 Aplicação final da Metodologia 7Cs na disciplina Algoritmos	113
6 MATERIAL DE ORIENTAÇÕES DA METODOLOGIA 7Cs	127
6. 1 POR QUE CRIAR UM MATERIAL DE ORIENTAÇÕES?.....	127
6. 2 REQUISITOS PARA A CONCEPÇÃO DO MATERIAL	128
6. 3 ORGANIZAÇÃO DO MATERIAL.....	130
6. 4 PROJETO GRÁFICO.....	132
6.5 GRUPO NO FACEBOOK	136
CONSIDERAÇÕES FINAIS.....	138
REFERÊNCIAS BIBLIOGRÁFICAS.....	142
APÊNDICE I – ROTEIRO DE OBSERVAÇÃO	146
APÊNDICE II – LISTAS DE EXERCÍCIOS - ESTRUTURA REPETIÇÃO (PRIMEIRO TESTE)	148
APÊNDICE III – LISTA DE EXERCÍCIO NA ESTRUTURA VETOR (NOVO TESTE) ..	154

APÊNDICE IV – LISTAS DE EXERCÍCIOS - ESTRUTURA MATRIZES (TESTE FINAL)	
.....	155
ANEXO I – QUESTÕES DA FASE APERFEIÇOAMENTO NO <i>KAHOOT!</i>	160

INTRODUÇÃO

Muitos alunos chegam à universidade com insuficiência em sua formação básica referentes à escrita, à interpretação textual, aos conceitos matemáticos, ao raciocínio lógico, à capacidade de abstração e à solução de problemas, habilidades consideradas importantes para a disciplina de algoritmos, normalmente ofertada no primeiro ano de formação de quem cursa graduação voltada para a área de Tecnologia de Informação (TI). Tais insuficiências, vindas de uma educação básica precária, têm sido apontadas como fatores impactantes relacionados ao cancelamento ou abandono da disciplina de algoritmos, pelas dificuldades dos alunos na compreensão e abstração do conhecimento. Giraffa e Mora (2013) enfatizam que a evasão em disciplinas dos níveis iniciais dos cursos de Computação não se trata de um tema novo, mas algo que muito vem sendo estudado, tanto em nível nacional quanto internacional.

Em cursos da área da computação, a disciplina de algoritmos é considerada como um grande desafio aos alunos. Muitas das expectativas dos alunos sobre a disciplina e o próprio curso de graduação são quebradas ao se depararem com o novo conhecimento, o qual não fez parte da sua formação inicial e que exige conhecimentos específicos e integrados. Dessa maneira, para aqueles que possuem lacunas na formação, atender essas demandas não é um processo fácil. Assim, muitos alunos ficam desmotivados e não se sentem capazes de desenvolver algoritmos com eficiência. Segundo Branco e Schuvartz (2007, p. 521):

Os cursos da área de computação e informática enfrentam um grande problema com as disciplinas de introdução à programação de computadores, as quais visam ensinar como utilizar o computador para solucionar problemas. Acadêmicos iniciantes, ao se depararem com a disciplina, sentem-se incapazes de programar, devido ao conjunto de habilidades que a programação exige como capacidade para solucionar problemas, raciocínio lógico, habilidade matemática, capacidade de abstração, entre outras.

Levamos em consideração que, diferentemente de alguns conteúdos, como os relacionados à Biologia, à Matemática e ao Português, por exemplo, a disciplina de algoritmos aborda um conhecimento difícil de se fazer presente na grade curricular do ensino básico, principalmente em escolas públicas. Dessa forma, muitos alunos que saem diretamente do ensino médio e ingressam nos cursos de computação, nunca

ouviram falar sobre o que é, e do que se trata a construção de um algoritmo, dificultando seu desempenho na disciplina. Os alunos percebem a disciplina e o conteúdo visto como algo distante da sua realidade e de difícil compreensão.

Um fato observado é que a maior parte dos alunos são egressos do ensino médio regular, que não tiveram base técnica de lógica ou algoritmos, o que contribui de forma direta para o aumento da dificuldade de aprendizado. Os alunos oriundos do segundo grau que fizeram o curso Técnico em Informática sempre apresentam maior facilidade nos conteúdos, o que direciona os índices de aprovação (VIEIRA; LIMA JUNIOR, VIEIRA, 2015, p. 13).

Petry (2005), por exemplo, relata que as dificuldades inerentes aos alunos quando não solucionadas em tempo hábil, trarão muitos prejuízos em sua formação acadêmica e para os docentes, que precisam cumprir o que é exigido no currículo. Esse fator ocasiona problemas futuros nas exigências de competências e habilidades necessárias em outras disciplinas durante o curso. A autora aponta que uma das dificuldades principais na disciplina de algoritmos em cursos da Computação é a compreensão dos conteúdos básicos e a aplicação da lógica de programação na construção de algoritmos para resolução de problemas. Podemos dizer que esse aspecto está diretamente voltado à encadeação de ideias e pensamentos para construção e solução de um problema por computador.

Outra implicação nas dificuldades apresentadas pelos discentes na disciplina é referente à forma tradicional de ensino, com o uso de práticas engessadas, que minimamente leva em consideração a individualidade e ritmo de aprendizagem. Pouco se trabalha na perspectiva de desenvolver habilidades e competências, tais como: a compreensão, a autonomia, a interpretação e a resolução de problemas. Esse fato pode ser alterado com a inserção de métodos adequados para construção do conhecimento dos alunos, por meio de metodologias que proporcionem e facilitem a compreensão dos conteúdos da disciplina.

O ensino de Algoritmos na maioria das Instituições se dá através de aulas convencionais e essas não privilegiam o que se faz necessário para a aprendizagem desse conteúdo. São necessárias ações pedagógicas que contemplem esse processo, ou seja, são necessárias ações pedagógicas voltadas à aprendizagem para a Resolução de Problemas que permitem desenvolver capacidades como atenção, concentração, pensamento criativo e outras habilidades perceptuais

psicomotoras indispensáveis para agilizar o raciocínio (RAPKIEWICZ *et al.*, 2006, p. 04).

Neste contexto, é importante que o professor busque estratégias e metodologias de ensino que possam auxiliá-lo e contribuir com a aprendizagem dos estudantes, em busca de facilitar a compreensão sobre os conteúdos de algoritmos. De acordo com Berssanette (2016), uma das dificuldades de ensino da disciplina está justamente relacionada à ausência de metodologias adequadas, pois geralmente é ministrada de maneira tradicional, apenas com a teorização.

Com base no exposto pretendemos responder a seguinte questão-foco:

O desenvolvimento de uma metodologia que sistematiza ações lógicas em processos de ensino e aprendizagem de algoritmos, pode contribuir para compreensão dos conteúdos básicos, minimizando dificuldades dos discentes de graduação de cursos que contemplam disciplinas introdutórias a programação?

Diante a questão-foco também determinamos como objetivo geral:

Conceber, desenvolver e testar Metodologia 7Cs (Compreender, Conceber, Completar, Compatibilizar, Corrigir, Construir e Contribuir), como proposta de ensino, para facilitar o processo de aprendizagem dos conteúdos básicos de algoritmos, com o intuito de minimizar dificuldades dos discentes de graduação de cursos que contemplam disciplinas introdutórias a programação.

Para alcançar o objetivo geral, foram elencados os seguintes objetivos específicos:

- Identificar as dificuldades de ensino e de aprendizagem na disciplina algoritmos;
- Conceber e desenvolver a Metodologia 7Cs;
- Testar a Metodologia 7Cs por meio de um curso piloto;
- Aplicar a Metodologia 7Cs em contexto real de sala de aula em uma disciplina da graduação.

Propomos como produto para esta pesquisa, uma metodologia de ensino e aprendizagem, para a prática dos conteúdos básicos de algoritmos, destinada à auxiliar professores em minimizar as dificuldades dos alunos, em disciplinas introdutórias de programação. Para consolidar a concepção da Metodologia 7Cs, realizamos um teste piloto e aplicamos no curso de Ciência da Computação da UFPA, no Instituto de Ciências Exatas e Naturais (ICEN) em laboratório de informática que disponibilizou uma estrutura propícia ao desenvolvimento da proposta.

JUSTIFICATIVA

Neste tópico discorreremos sobre as inquietações que justificaram e motivaram a construção da proposta de concepção da Metodologia 7Cs e o porquê de esta ser relevante e poder ser considerada como criativa e inovadora. Nos três subtópicos seguintes utilizamos a escrita na primeira pessoa do singular, por se tratar de experiências pessoais do mestrando, autor desta pesquisa.

Primeiras inquietações e motivação

Na chegada ao ensino médio, o sonho de ingressar em uma universidade floresceu e se tornou um objetivo de vida. Ao finalizar o ensino básico em 2011, decidi prestar vestibular para Universidade Federal Rural da Amazônia (UFRA) para ingressar no curso de Licenciatura em Computação, e aos 17 anos, aquele menino vindo do interior, com grandes expectativas conseguiu realizar seu sonho de passar no Vestibular.

Ao começar a cursar a graduação pude perceber as inúmeras lacunas deixadas pela minha formação na educação básica, e que necessitavam ser preenchidas, visto que muitas habilidades e competências que deveriam ter sido desenvolvidas e aprendidas na base se perderam ao longo do processo de aprendizagem.

A falta de desenvolvimento de tais habilidades foram sentidas principalmente no segundo semestre do curso de graduação, na disciplina Algoritmos, em que minhas expectativas iniciais foram estremecidas, devido ao alto nível de complexidade e exigência impostos. Uma das principais exigências era o uso do raciocínio lógico e a interpretação dos comandos, de maneira a organizar uma linha de pensamento de

forma a solucionar determinados problemas com a construção de algoritmos. Tal dificuldade foi vivenciada pela maioria dos alunos da minha turma.

Algo a se destacar é que a professora da disciplina se preocupava em ensinar de maneira que todos pudessem aprender, pois repetia quantas vezes fossem necessários alguns conteúdos. Porém, era praticamente impossível em uma turma com mais 30 alunos que ela conseguisse sanar as dúvidas de todos, pois ainda tinha que dar conta de todo o conteúdo do semestre.

Após o término da disciplina tive a oportunidade, ainda na graduação, de ingressar no Programa Institucional de Bolsas de Iniciação à Docência (PIBID) e vivenciar grandes experiências no projeto intitulado “Desenvolvimento do Raciocínio Lógico e Algoritmos” durante o período de 2013 a 2016, em que ministrava aulas de raciocínio lógico e algoritmos para alunos do ensino médio e fundamental. Foi um grande desafio, pois ensinar um conteúdo tão complexo no qual enfrentei grandes dificuldades inicialmente, não foi uma tarefa fácil.

O Programa Institucional de Bolsas de Iniciação à Docência (PIBID), destinado aos discentes de Licenciatura em Computação da Universidade Federal Rural da Amazônia (UFRA), proporciona aos bolsistas a oportunidade de desenvolverem a prática docente, uma experiência de suma importância por serem futuros profissionais da educação em computação (LIMA; SOUSA, 2015, p. 1291).

A coordenadora do projeto sempre demonstrou interesse por novas formas de atuação, que pudessem ajudar os bolsistas a desenvolver suas próprias estratégias de ensino. Assim, a partir das experiências vivenciadas durante o projeto, consegui identificar inúmeras possibilidades de ensino para a disciplina de algoritmos, ao demonstrar junto aos colegas bolsistas que esta poderia ser ministrada de forma diferenciada, mesmo no ensino superior. Tal experiência foi compartilhada mediante a publicação de artigos em Congressos de informática e Educação.

Outro aspecto interessante e que também merece destaque se refere aos exercícios aplicados durante o curso de algoritmos realizado na educação básica, visto que os discentes da graduação sentiram maior dificuldade em compreender algumas das questões, do que os alunos do ensino básico. Esse fato ocorreu devido à metodologia aplicada pelos bolsistas em utilizar dinâmicas, desafios e exercícios por meio de

práticas lúdicas, voltadas ao processo de aprendizagem dos conteúdos básicos de algoritmos no curso.

Diante do exposto, ficou claro que as estratégias utilizadas por meio da ludicidade e estímulo ao raciocínio lógico, tornaram o ensino da disciplina diferenciado, dinâmico e atrativo, o que fez grande diferença na construção do conhecimento acerca dos conteúdos ministrados.

A partir dessa experiência percebi que poderiam existir outras formas de ensinar e contribuir para minimizar as dificuldades encontradas na disciplina de algoritmos nos cursos de graduação, assim como as estratégias utilizadas no ensino básico. Esse fato serviu de inspiração para o meu desejo em ingressar no mestrado com um projeto voltado a contribuir para minimizar as dificuldades sentidas por alunos de graduação na compreensão dos conteúdos da disciplina Algoritmos, pois senti necessidade de transformar essas inquietações em novas possibilidades.

Transformação das inquietações em busca de mudanças

A experiência que pude vivenciar como discente no curso de Licenciatura em Computação da UFRA, trouxe à tona antigas inquietações sobre a disciplina de algoritmos, pois tive a consciência de que muitos alunos passavam pela mesma experiência e enfrentavam as mesmas dificuldades que sentia. Dessa maneira, me senti desafiado a conceber um produto educacional que pudesse facilitar o processo de aprendizagem na disciplina de algoritmos, isto é, uma proposta diferenciada de ensino.

Tais inquietações, que emergiram das dificuldades enfrentadas na disciplina impulsionaram a busca por outras maneiras de ensinar algoritmos na graduação, e suscitaram motivações para a construção do anteprojeto para concorrer a uma vaga no Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior (PPGCIMES). A escolha de participar da seleção do PPGCIMES se fez por ser o único mestrado que contemplava de forma mais concreta o projeto que eu pretendia desenvolver.

Durante a trajetória nas disciplinas obrigatórias no PPGCIMES, grandes transformações ocorreram. Muitas das estratégias que havia pensado em realizar inicialmente foram desconstruídas e reconstruídas por meio de reflexões e apropriação dos conhecimentos sobre criatividade e inovação que o mestrado proporcionou.

Aconteceram mudanças significativas na concepção do produto, que passou por um processo de reconstrução, uma quebra de realidade, de conhecimentos pré-existentes e engessados, para uma nova forma de ver o ensino, que me possibilitasse conceber um produto educacional criativo e inovador, para alcançar aos objetivos desejados.

Por essa razão, a partir das inquietações suscitadas e das experiências vivenciadas no mestrado, compreendi a importância da criação de metodologias que possam facilitar a aprendizagem dos alunos sobre os conteúdos ministrados na disciplina de algoritmos, partindo não só das minhas premissas individuais, mas também do diálogo com os indivíduos envolvidos e do entendimento do contexto vivenciado.

Processo e escolha do produto

“Quem disse que seria fácil?”. Não foi nada fácil chegar à escolha do que realmente produzir e desenvolver como produto educacional. Foi um processo demorado, de altos e baixos, mas que proporcionou grandes aprendizados. Aprendizados estes que ainda estão em constante construção. No percurso realizado, várias vezes vinham a mente as fases de mudanças e descobertas sobre o que pretendia elaborar como produto criativo e inovador. Essas idas e vindas em busca de uma definição para o produto a ser realizado, possibilitou percorrer um processo de constante mudança, no qual foi possível visualizar as fases relatadas por Kneller (1978) sobre o ato de criar: *primeira apreensão, preparação, incubação, iluminação e verificação*.

O *insight* para a concepção do produto surgiu com o desejo de minimizar as dificuldades na disciplina de algoritmos. Essa ideia inicial se relaciona ao que Kneller (1978) aponta como a primeira fase, a *primeira apreensão*. Esta fase se inicia a partir de algum problema a ser resolvido, no caso o problema vivenciado por alunos na disciplina de algoritmos. Ao partir da premissa de que um dos fatores que provocam as dificuldades vivenciadas pelos alunos na disciplina acontece devido à falta de conhecimento prévio sobre os conteúdos trabalhados, como também por causa das inquietações trazidas por mim desde a graduação, surgiu a ideia da criação de um curso prévio para introduzir os conteúdos, como intuito de elaborar estratégias de aprendizagem que pudessem minimizar as dificuldades e contribuir para um bom

rendimento na disciplina. O curso seria a minha proposta inicial para ingressar no PPGCIMES, como produto da dissertação, porém não foi executado.

Ingressar no mestrado, me fez perceber que existiam algumas lacunas que precisavam ser preenchidas para concepção do produto, pois não se tratava de um processo simples, mas que exigia uma grande preparação e investigação sobre o que pretendia realizar. Essa fase é chamada de *preparação* e se “constitui em uma rigorosa investigação das potencialidades da ideia germinal” (KNELLER, 1978, p.63), a qual se estendeu por um longo período e veio se consolidar a partir da observação participante realizada no primeiro semestre de 2018 na disciplina Algoritmos ofertada no Curso de Bacharelado em Ciência da Computação da UFPA. Nesse período foi possível verificar algumas das reais necessidades e as possíveis soluções para o problema dos alunos na disciplina, o que suscitou o seguinte questionamento: “A realização do curso seria suficiente para alcançar os objetivos pretendidos?”.

Com base no referencial teórico estudado, além da observação participante e as orientações recebidas cheguei à conclusão que, junto aos orientadores, deveríamos pensar em uma nova estratégia para que o produto pudesse ser mais coerente e trazer contribuições para compreensão dos conteúdos da disciplina, considerando as dificuldades na aprendizagem, habilidades e competências exigidas. Dessa maneira passamos para a fase de *incubação* que, de acordo com Kneller (1978), que não se distancia totalmente da fase de *preparação*. Para ele, o período de preparação é algo consciente e seguido também pelo inconsciente. A fase de incubação se constitui como uma fuga do produto, para verificar novas possibilidades, guardando as ideias inicialmente pensadas.

No segundo semestre de 2018 realizei o Estágio Supervisionado, componente curricular obrigatório no Mestrado, em uma turma de Engenharia Elétrica do Instituto de Tecnologia (ITEC) da UFPA, que possui a disciplina Programação Estruturada de Computadores, em sua grade curricular. Para realizar o Estágio, precisei elaborar uma estratégia de ensino que contemplasse pelo menos a unidade da disciplina. Então, propus uma sequência lógica de ações para aprendizagem de algoritmos, de maneira bem simples, para a unidade de Estrutura de Seleção. A partir dessa experiência, outras ideias surgiram, até que concluímos que o produto, mais importante do que um curso, seria desenvolver uma metodologia de ensino e de aprendizagem que pudesse

sistematizar uma sequência lógica de ações, para melhor compreensão dos conteúdos básicos de algoritmos.

Diante do exposto, realizei levantamento bibliográfico sobre a concepção de metodologias voltadas para disciplina de algoritmos. Nessa busca, durante o processo de orientação encontrei-me respaldo, para a ideia de concepção de uma metodologia, nas palavras de Campos (2009, 2010, 2011) que se relacionava muito bem com a proposta a ser desenvolvida, visto que ele concebeu uma metodologia que, apesar de não nomear dessa forma, apresenta uma sequência lógica de ações para compreensão da disciplina de Lógica de Programação, que equivale a de algoritmos. Porém, até aquele momento, ainda não havíamos chegado à conclusão de como seria estruturada ou desenvolvida a metodologia. Em uma conversa informal com a coorientadora do trabalho no corredor do Instituto de Ciências Exatas e Naturais (ICEN), após a saída do Estágio Supervisionado, vivenciamos a fase de *iluminação*, e criamos a Metodologia 5C, composta das seguintes dimensões: Compreender, Compor, (Re) Compor, Completar e Corrigir, com a proposta de apresentar uma forma de exercitar os conteúdos da disciplina de algoritmos. O momento de iluminação é um clímax do processo de criação em que o criador percebe a solução do seu problema, o pensamento que completa todas suas ideias (KNELLER, 1978).

Após essa decisão, comecei a estruturar a metodologia. Durante o processo de estruturação, ocorreu a fase de *verificação*. Nessa fase o criador verifica e revisa, o que é válido e o que não é válido, pois a iluminação é vista como notoriamente falha (KNELLER, 1978). A partir da verificação, estudos sobre metodologias criadas para a disciplina de algoritmos, levantamentos de dados da observação e Estágio Supervisionado, inspirados em Campos (2009), alguns pontos foram melhorados, modificados e acrescentados, que culminaram, na constituição da Metodologia 7Cs, agora com sete dimensões: Compreender, Conceber, Completar, Compatibilizar, Corrigir, Construir e Colaborar.

Para verificar de forma mais concreta a viabilidade de aplicação da Metodologia 7Cs no contexto de sala de aula, realizei um teste piloto no curso de nivelamento de algoritmos na UFPA, para calouros dos cursos de graduação em Bacharelado em Ciências da Computação e Sistemas de Informação no período de 12/03 a 15/03 de 2019, o qual será descrito com mais detalhes em outro capítulo. Nele consegui ver outras estratégias e possibilidades de uso da metodologia. Após o teste passei pelo

Exame de Qualificação do Mestrado, momento no qual a banca apresentou valiosas sugestões, como o acréscimo de uma nova dimensão, Criar. Por fim, apliquei no primeiro semestre de 2019 a metodologia na disciplina Algoritmos do Curso de Graduação em Bacharelado em Ciências da Computação, também na UFPA.

Com base nas sugestões e análises dos resultados que obtive ao aplicar a metodologia, desenvolvi o que consideramos como versão final da Metodologia 7Cs com as sete dimensões: Compreender, Conceber, Completar, Compatibilizar, Corrigir, Construir e Criar. Ao pensar em tornar aprendizagem dos alunos mais significativa, estruturamos a metodologia em três fases Descoberta, Aperfeiçoamento e Consolidação, baseadas na Teoria de Aprendizagem Significativa de Ausubel (2000). É válido ressaltar que a dimensão Colaborar não foi excluída, ela se tornou uma ação transversal a metodologia, em que, incentivamos a colaboração entre os alunos, durante a execução de suas fases.

Inovador e Criativo

Como já citado, o PPGCIMES proporcionou importantes aprendizados e reflexões acerca de inovação e criatividade, ao desconstruir e reconstruir conhecimentos consolidados sobre o que é ser criativo e inovador. Desenvolver e conceber um produto que seja criativo e inovador não é tarefa fácil, pois requer considerar o que é externo e não apenas as particularidades intrínsecas às vontades pessoais. Para Schwengber, Silveira e Ribeiro (2015, p. 6), os “objetos inovadores e criativos não emanam a partir de iluminações isoladas da mente humana”, é preciso perceber o que está além das premissas individuais, levando em consideração as necessidades do público ao qual o produto se destina. A partir desse princípio, buscamos primeiramente verificar as reais necessidades do público para o qual foi pensado a concepção e desenvolvimento da Metodologia 7Cs, considerando as percepções das dificuldades apresentadas pelos alunos durante a observação na disciplina de algoritmos.

Neste sentido, Masetto (2012, p. 16) relata que as inovações educacionais não se apresentam neutras, mas “sempre surgem como resultado de um contexto social, de determinada concepção de educação e como resposta a necessidades emergentes para as quais os paradigmas atuais já não oferecem encaminhamentos aceitáveis”.

Acreditamos que a Metodologia 7Cs é inovadora, porque está sendo pensada e desenvolvida com intuito de contribuir com a realidade a partir do contexto vivenciado, propondo uma forma de aprendizagem e prática da disciplina de algoritmos de maneira significativa.

O que estamos propondo é uma maneira diferenciada que contrapõe o ensino apenas com teorização, que não leva em consideração a individualidade e particularidades do contexto vivenciado. De acordo com Torres (2005, p.38), “inovação é um processo dinâmico de mudanças específicas que tem como resultado o crescimento pessoal, institucional e social. Na inovação destaca-se o lado institucional e social da criatividade”. Assim, tudo o que pensamos em conceber como algo inovador, também equivale para a dimensão da criatividade.

Ao pensar em conceber um processo e produto inovador e com características criativas, devemos nos atentar as habilidades associadas à criatividade. Conforme Alencar (2003):

[...] para que um produto criativo seja gerado, conhecimento e técnica não são suficientes. É necessário que o indivíduo desenvolva, também, habilidades associadas à criatividade, tais como flexibilidade, originalidade, sensibilidade a problemas e imaginação (ALENCAR, 2003, p. 48).

Com base nessa afirmativa, buscamos propor um produto de cunho criativo e inovador. Para avaliar se a Metodologia 7Cs tem em sua concepção potenciais e características criativas e inovadoras, elaboramos um quadro (Quadro xx) contendo dimensões e critérios que julgamos termos alcançado. As dimensões e critérios, descritos a seguir, tiveram como base a atividade “Processos e produtos educacionais criativos” da disciplina Criatividade do PPGCIMES e nos autores Besemer & Taefingea (1981), Wechsler (1993), Horn & Salvendy (2006), Morais (2009), Schwengber e, Silveira e Ribeiro (2015):

- **Dimensão “Novidade do Produto”:** cada produto criativo desenvolvido deve apresentar algum grau de novidade em relação a novas técnicas, novos conceitos, novidade que traz a extensão de novos processos criativos, entre outros.

Critérios desta dimensão:

- **Originalidade:** contempla uma nova combinação de elementos existentes, assim como a infrequência de um produto.
 - **Germinabilidade:** é basicamente a possibilidade de geração de novos produtos a partir de outro.
- **Dimensão “Transformação”:** pode ser definida à medida que o produto força uma nova maneira de olhar o contexto vivenciado.

Critérios desta dimensão:

- **Contextualização:** é a relação do produto com o meio no qual foi criado e a interação entre o contexto, produto e pessoa.
 - **Impacto social:** quando o produto criativo gera subsídios básicos e necessários à essência humana, a partir de uma observação do passado e presente para desenvolvimento cultural, pessoal e social do futuro.
- **Dimensão “Aplicabilidade”:** mede o quão importante é o produto para o usuário que dele fará uso e, conseqüentemente, sua aplicação no contexto para o qual foi desenvolvido.

Critérios desta dimensão:

- **Flexibilidade:** pode ser entendida como a possível mudança das soluções propostas, que se adapta a outros contextos e situações.
 - **Adequação:** satisfazer uma necessidade ou na realização de uma função desejada, do produto.
 - **Expressivo:** diz respeito à facilidade com a qual a pessoa interage com o invento.
- **Dimensão “Estética”:** está relacionada à observação do apelo estético, no que tange quanto o produto é externamente atraente ao usuário e ao mercado.

Critérios desta dimensão:

- **Atraente:** entendido como algo que atrai a atenção do espectador ou observador.

- **Bem trabalhada:** refere-se à quantidade de esforço despendido na produção de produtos criativos e o cuidado com que ideias são desenvolvidas.

Com base nas dimensões e critérios descritos, apresentamos no Quadro 1, as dimensões e critérios que consideramos, ter na concepção da Metodologia 7Cs.

Quadro 1 - Criatividade do produto

DIMENSÃO	CRITÉRIOS	METODOLOGIA 7Cs
Aplicabilidade	Flexibilidade	A Metodologia 7Cs pode ser adaptada a outros contextos em diferentes linguagens de programação.
	Adequação	A proposta da Metodologia 7Cs surgiu a partir do contexto e das dificuldades observadas na disciplina de algoritmos e na literatura.
	Expressivo	A Metodologia 7Cs não necessita de aparatos tecnológicos, o que a torna de fácil acesso e manipulação.
Estética	Atratividade	A proposta da Metodologia 7Cs é trabalhar com atividades por meio de situações problemas relacionados a situações cotidianas, em busca de motivar os alunos.
	Bem trabalhada	A Metodologia 7Cs passará por um processo de 6 etapas na qual será fundamentada, verificada, testada e validada, para que possa ser consolidada.
Novidade do Produto	Originalidade	A originalidade da Metodologia 7Cs está na sua própria concepção, visto que estamos em processo de concepção e desenvolvimento de uma nova forma de abordagem e aprendizagem dos conteúdos básicos de algoritmos.
	Germinabilidade	A partir da Metodologia 7Cs poderão ser criadas novas metodologias de ensino.
Transformação	Contextualização	A Metodologia 7Cs foi pensada a partir da observação na disciplina de algoritmos, e leva em consideração o contexto, as dificuldades e necessidades dos alunos.
	Impacto social	A Metodologia 7Cs busca contribuir para mudança do cenário de altos índices de reprovações na disciplina de algoritmos, muito presente nos cursos de Computação.

Fonte: Tabela adaptada da atividade Processos e Produtos Criativos, utilizado como requisito de avaliação na disciplina Criatividade, ministrada no PPGCIMES - Turma 2018.

A relevância da proposta consiste em conceber e desenvolver uma metodologia de ensino e aprendizagem que busca organizar uma sequência lógica para construção do conhecimento e compreensão de conteúdos básicos de algoritmos. Para isso, utilizamos como suporte a Teoria de Aprendizagem Significativa de Ausubel (2000), em

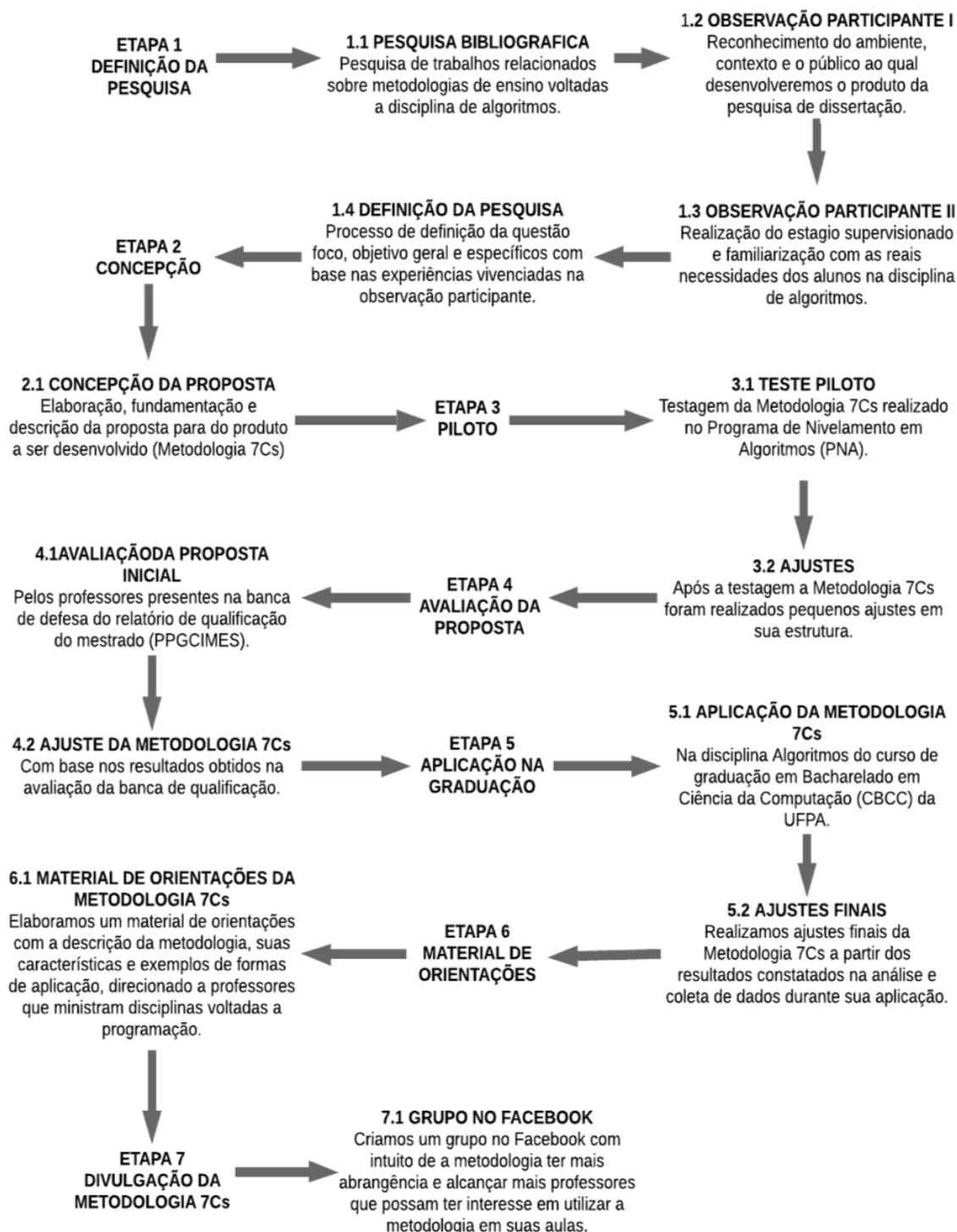
busca de potencializar o desenvolvimento de habilidades e competências, com foco em minimizar dificuldades, muito frequentes nos cursos de Computação.

FLUXO METODOLÓGICO DA PESQUISA

A concepção e desenvolvimento da Metodologia 7Cs voltada para a aprendizagem na disciplina de algoritmos, foi realizada empregando a abordagem de pesquisa quali-quantitativa. O modelo misto de pesquisa começa quando o pesquisador define as categorias do estudo, e consiste em querer ir além de conhecimentos predeterminados e considerar o que acredita ser importante para sua pesquisa, mesmo que com o tempo isso modifique as categorias e ações determinadas inicialmente (LAVILLE; DIONNE, 1999). Nesse sentido, desenvolvemos a metodologia proposta, com o intuito de atender as reais necessidades do público para quem ele se destina.

O processo de desenvolvimento da Metodologia 7Cs ocorreu em sete etapas a saber: definição da pesquisa, concepção da proposta, testagem piloto, avaliação da proposta, aplicação na graduação, concepção de um material de orientações da metodologia e meio de divulgação da metodologia. A seguir apresentaremos na Figura 1 os procedimentos metodológicos que seguimos para a realização da nossa proposta.

Figura 1 - Procedimentos metodológicos para concepção da proposta de ensino no Metodologia 7Cs



Fonte: Elaborado pelo autor.

Na dissertação, apresentamos a forma como alcançamos algumas das etapas apresentadas na Figura 1. Todas elas foram importantes, para que pudéssemos desenvolver e consolidar a proposta de concepção da Metodologia 7Cs, ao possibilitar,

por exemplo, o levantamento de requisitos importantes de serem considerados, na constituição de uma metodologia de ensino e aprendizagem.

ESTRUTURA DA DISSERTAÇÃO

Para apresentação desta dissertação, utilizamos a seguinte estrutura, a partir da introdução. No **Capítulo 1** apresentamos: (i) as dificuldades dos alunos na compreensão dos conteúdos básicos de algoritmos, assim como os fatores que influenciam no baixo desempenho; (ii) as dificuldades dos professores no processo de ensino em disciplinas de algoritmos; e (iii) a importância da prática para o processo de aprendizagem e o rendimento dos alunos.

No **Capítulo 2**, por sua vez apresentamos a base teórica da pesquisa, em que, abordamos: (i) a Teoria de Aprendizagem Significativa de Ausubel (2000), a qual fundamenta e estrutura a Metodologia 7Cs e, (ii) trabalhos de dissertações correlatos ao nosso, que utilizaram os princípios de aprendizagem significativa para o ensino de algoritmos.

No **Capítulo 3** apresentamos: (i) trabalhos relacionados à concepção de estratégias e metodologias para o ensino de algoritmos; (ii) de maneira sucinta, o trabalho que nos inspirou a desenvolver a Metodologia 7Cs; (iii) relatos de observações participantes realizadas em duas diferentes turmas de graduação em disciplinas introdutórias a programação; (iv) relato de uma pesquisa exploratória realizada na graduação do Curso de Bacharelado em Ciência da Computação (CBCC) da Universidade Federal do Pará; e (v) os requisitos que subsidiaram a constituição e desenvolvimento da Metodologia 7Cs.

Já no **Capítulo 4** descrevemos: (i) a Metodologia 7Cs, apresentando suas dimensões, seus objetivos, as habilidades e competências e seus pré-requisitos; (ii) as fases da metodologia e a relação delas com a teoria de aprendizagem significativa; e (iii) a classificação da categoria da metodologia, enquanto um produto educacional.

No **Capítulo 5** manifestamos nos impressões sobre o uso da Metodologia 7Cs, e levantamentos sobre a percepção dos alunos em relação ao desenvolvimento de habilidades e competências, em que apresentamos sua aplicação tanto em um curso de nivelamento de conceitos básicos de algoritmos quanto em um contexto real de sala de aula na graduação no CBCC da UFPA.

No **Capítulo 6** discriminamos sobre a criação do material de orientações para o uso da metodologia 7Cs, apresentando os objetivos do material concebidos e todas as suas características.

Por último apresentamos as **Considerações Finais** em relação aos objetivos definidos para a pesquisa, assim como buscamos mostrar indícios de que a metodologia pode contribuir com ensino e aprendizagem de algoritmos, e as possibilidades de trabalhos futuros para continuidade e evolução da Metodologia 7Cs.

1 ENSINO E APRENDIZAGEM DE ALGORITMOS E SEUS DESAFIOS

O presente capítulo apresenta algumas dificuldades dos alunos na disciplina de algoritmos, bem como fatores que influenciam no seu desempenho, tais como: a baixa motivação, o primeiro contato com o conteúdo, a falta de desenvolvimento de habilidades e competências na educação básica, as dificuldades dos professores no ensino da disciplina e a importância da prática no processo de aprendizagem.

1.1 DIFICULDADES NA DISCIPLINA DE ALGORITMOS NO ENSINO SUPERIOR

A disciplina de algoritmos nos cursos de computação é considerada uma das mais importantes, pois serve de base para demais disciplinas sendo geralmente ofertada no primeiro semestre do curso. Para Barcelos *et al.* (2009) o estudo de algoritmos é uma área fundamental no campo da Computação, uma vez que o conhecimento e a compreensão sobre algoritmos permitirão a aprendizagem nas linguagens de programação, que em sua essência possuem estratégias de solução de problemas.

É notório que em todos os cursos de tecnologia ou graduação da área de Computação, a disciplina de Algoritmos constitui eixo básico essencial na formação dos alunos. Por ser o primeiro contato direto dos discentes com ambientes que trabalham os conceitos de raciocínio lógico, matemática e programação, pode-se evidenciar que em todos os cursos, tanto de instituições públicas quanto privadas, a disciplina se faz presente desde o primeiro ano da formação do aluno (VIEIRA; LIMA JUNIOR; VIEIRA, 2015, p. 6).

A baixa motivação é um dos pontos citados como um dos responsáveis pela evasão na disciplina. Muitas vezes o déficit dos alunos não é observado pelo professor, por apenas se prender ao método de ensino instrucionista e tradicional, que não leva em consideração as dificuldades e limitações sobre as habilidades e competências exigidas pelas disciplinas. São várias as dificuldades enfrentadas pelos alunos nessa, o que acaba por conduzir ao insucesso acadêmico (AMBRÓSIO, 2011). Em seu estudo de dissertação, Berssanette (2016) realizou uma revisão de literatura sobre as dificuldades inerentes aos alunos sintetizadas conforme o Quadro 2, a seguir.

Quadro 2 - Síntese das principais dificuldades de aprendizagem discutidas por alunos de programação, mencionadas na literatura (Continua...)

DIFICULDADES	AUTORES
Dificuldade em leitura e interpretação dos enunciados	(DEREMER, 1993) (WILSON; SHROCK, 2001) (FALKEMBACH; AMORETTI; et al., 2003) (DELGADO et al., 2005) (GOMES; MENDES, 2007)
Dificuldade no desenvolvimento do raciocínio lógico	(DEREMER, 1993) (WILSON; SHROCK, 2001) (RAABE; SILVA, 2005) (PEREIRA JÚNIOR et al., 2005) (SANTOS; COSTA, 2006) (GOMES; MENDES, 2007)
Poucas habilidades na resolução de problemas	(FALKEMBACH; AMORETTI; et al., 2003) (GOMES, A.; MENDES, 2007) (MARTINS; MENDES; FIGUEIREDO, 2010)
Baixa capacidade de abstração	(PEREIRA JÚNIOR; RAPKIEWICZ, 2004) (RAABE; SILVA, 2005) (HABERMAN; MULLER, 2008) (PIVA JÚNIOR; FREITAS, 2010) (MARTINS; MENDES; FIGUEIREDO, 2010)
Baixo nível de conhecimento em matemática	(HENDERSON, 1987) (DEREMER, 1993) (WILSON; SHROCK, 2001) (KOLIVER; DORNELES; CASA, 2004) (RAABE; SILVA, 2005) (GOMES; MENDES, 2007) (MOTA et al., 2009)
Hábitos de estudo equivocados	(BIGGS, 1999) (JENKINS, 2001) (RODRIGUES, 2002) (PEREIRA JÚNIOR; RAPKIEWICZ, 2004) (KOLIVER; DORNELES; CASA, 2004)
Estilos de aprendizagem diferentes	(BIGGS, 1999) (PEREIRA JÚNIOR et al., 2005) (CÂNDIDA; MARCELINO; MENDES, 2007) (CASPERSEN; BENNEDSEN, 2007)
Pouca motivação	(JENKINS, 2001) (RODRIGUES, 2002) (RODRIGUES JÚNIOR, 2004) (GOMES; MENDES, 2007) (MARTINS; MENDES; FIGUEIREDO, 2010)

Fonte: Adaptado de Berssanette (2016, p.27).

Quadro 2 – Síntese das principais dificuldades de aprendizagem discutidas por alunos de programação, mencionadas na literatura (Conclusão.)

DIFICULDADES	AUTORES
Problemas extraclases (vida pessoal), adaptação a nova vida acadêmica	(RODRIGUES, 2002) (PEREIRA JÚNIOR et al., 2005)
Pré-conceito atribuído as disciplinas de programação	(PEREIRA; JÚNIOR; RAPKIEWICZ, 2004)
Falta de persistência, ou pouco empenho	(JENKINS, 2001) (FORTE; GUZDIAL, 2005) (GOMES; MENDES, 2007)

Fonte: Adaptado de Berssanette (2016, p.27).

As dificuldades existentes na aprendizagem são perceptíveis, o que torna inviável no decorrer da disciplina suprir todas as necessidades individuais de cada aluno. Na disciplina de algoritmos, ainda no ensino tradicional, o professor é visto como a pessoa a qual deve acompanhar cada etapa dos alunos no desenvolvimento dos problemas propostos, mas na prática é muito difícil realizar o acompanhamento individual de todos (PETRY, 2005). Diante do contexto se faz necessário encontrar estratégias e métodos de ensino e aprendizagem que possam auxiliar e contribuir com a aprendizagem dos alunos, visto que “a apropriação ou não dos conceitos iniciais de programação tem relação direta com o desempenho do aluno no decorrer de todo o curso, já que disciplinas avançadas dependem fortemente desses conceitos” (ROCHA *et al*, 2010, p. 2).

É válido ressaltar que as dificuldades não são inerentes apenas aos alunos, os professores também possuem dificuldades no ensino da disciplina. Berssanette (2016) também realizou uma revisão bibliográfica e levantou dificuldades dos professores no ensino de algoritmos, sintetizadas no Quadro 3, a seguir.

Quadro 3 - Síntese das dificuldades de ensino de algoritmos por professores mencionadas na literatura

DIFICULDADES	AUTORES
Ausência de metodologias e/ou práticas de ensino adequadas	(JENKINS, 2001) (GIRAFFA; MARCZAK; ALMEIDA, 2003) (PIMENTEL; FRANÇA; OMAR, 2003) (RODRIGUES JÚNIOR, 2004) (SANTOS; COSTA, 2006) (CASPERSEN; BENNEDSEN, 2007) (GOMES; HENRIQUES; MENDES, 2008)
Restrição de tempo	(KOLIVER; DORNELES; CASA, 2004) (ROCHA et al., 2010)
Impossibilidade de acompanhamento individual da aprendizagem dos alunos	(JENKINS, 2001) (TOBAR et al., 2001) (BIGGS, 2003) (RAABE; SILVA, 2005) (ROCHA et al., 2010)
Ensino sem enfoque na resolução de problemas	(NOBRE; MENEZES, 2002) (KOLIVER; DORNELES; CASA, 2004) (GOMES; MENDES, 2007)
Abordagem pouco motivadora	(BORGES, 2000) (KOLIVER; DORNELES; CASA, 2004) (RODRIGUES JÚNIOR, 2004)
Heterogeneidade dos alunos	(ROBERTS, 1999) (ROBERTS, 2001) (KOLIVER; DORNELES; CASA, 2004)

Fonte: Adaptado de Berssanette (2016, p.25).

Dentre as dificuldades citadas, destacamos a falta de metodologias e/ou práticas adequadas que considerem as lacunas de aprendizagem pela ausência de pré-requisitos bem definidos na estrutura cognitiva dos alunos. Uma metodologia de ensino e aprendizagem deve ser capaz de proporcionar ao professor e aos alunos a facilitação na compreensão dos conteúdos.

1.2 A IMPORTÂNCIA DA PRÁTICA PARA A DISCIPLINA ALGORITMOS

A partir da análise das dificuldades dos alunos e professores na disciplina Algoritmos, constatamos que é preciso criar metodologias e práticas de ensino que possam contribuir no processo de aprendizagem. Por isso, ao pensar na concepção da

Metodologia 7Cs, buscamos identificar qual o principal elemento que influencia diretamente na compreensão dos conteúdos e na construção de algoritmos. Dessa forma, chegamos à conclusão que a metodologia deveria ser direcionada a parte prática da disciplina.

O processo para construção do conhecimento do aluno na aprendizagem de algoritmo é algo novo a sua estrutura cognitiva, o que representa muitas vezes um bloqueio ou resistência ao aprendizado dos conteúdos básicos da disciplina. De acordo com Araújo e Falckembach (2006, p. 01):

[...] resolver problemas inclui uma predisposição para o questionamento de quais os processos cognitivos que devem ser utilizados, a fim de gerar o conhecimento necessário à resolução do problema. Envolve as operações de pensar, de análise, de síntese e de avaliação.

Nesse sentido, tal resistência na aprendizagem e bloqueio, impede que os alunos consigam um bom desempenho. A disciplina de algoritmos é feita principalmente da prática de resolução de problemas por computador em que a praticar se torna um processo muito importante para que os alunos desenvolvam habilidades e competências necessárias a construção de algoritmos com eficiência.

O estudo de Timmermann (2015), realizou o acompanhamento de dois docentes que ministravam a disciplina de algoritmos, constatando que eles acreditavam ser a prática dos conhecimentos vivenciados algo fundamental para o processo de aprendizagem dos conteúdos ministrados. Esta autora ainda ressalta que “[...] exercitar, normalmente, demanda mera aplicação de técnicas [...]” (2015, p. 104), o que para ela é insuficiente para aprendizagem de algoritmos, pois pouco desenvolve funcionalidades inéditas nos exercícios propostos para os alunos das duas turmas observadas.

Estas tiveram, na maioria das vezes, suas resoluções explicadas previamente pelos professores, sendo quase que totalmente exemplificadas no quadro. Esta ação tornava a escrita desses algoritmos por parte dos alunos em aplicação de técnicas e não em elaboração de uma solução ou estratégia para resolver um problema (TIMMERMANN, 2015, p. 105).

A prática em disciplinas introdutórias à programação é crucial para concretizar os conhecimentos adquiridos durante o processo de aprendizagem, mas é importante

termos o cuidado para que a aprendizagem dos alunos não se resuma apenas, em aplicação de técnicas, que possam inibir seu processo criativo para solucionar problemas e a construção significativa do seu aprendizado sobre os conteúdos.

Aprender a desenvolver algoritmos é uma tarefa complexa, com inúmeras barreiras. Assim, é necessário propor soluções que ajudem ao professor e ao aluno a transporem os obstáculos que impedem a compreensão do conteúdo a ser ministrado. A partir do entendimento sobre a importância em praticar a resolução de problemas, pensamos na concepção e desenvolvimento de uma metodologia de ensino e aprendizagem para a prática de construção de algoritmos, que possa contribuir de maneira significativa com o aprendizado dos alunos, na busca de facilitar e minimizar dificuldades em compreender os conteúdos básicos da disciplina supracitada.

2 APRENDER DE FORMA SIGNIFICATIVA

Neste capítulo descrevemos a Teoria de Aprendizagem Significativa de Ausubel (2000), na perspectiva de Moreira (2011) e Moreira e Masani (1982), a qual fundamentou a concepção da Metodologia 7Cs. Apresentamos, também, trabalhos de dissertação relacionados à pesquisa desenvolvida, direcionadas a aplicação da Teoria de Aprendizagem Significativa de Ausubel (2000) aplicada em disciplinas de algoritmos (BERSSANETTE, 2016; BREZOLIN, 2015, BARBOSA, 2011).

2.1 APRENDIZAGEM SIGNIFICATIVA

O ensino superior exige um vasto conhecimento e dedicação do docente que nele está inserido. Igualmente ao qualquer outro formador, esse profissional necessita de saberes que possam auxiliá-lo nesta jornada de ensino, para que, dessa maneira, alcance meios de tornar a sala de aula eficiente e a aprendizagem efetiva do aluno, evitando um sistema somente conteudista.

David Paul Ausubel é considerado o representante do cognitivismo e se baseia na perspectiva cognitivista para explicação do processo de aprendizagem em que os conceitos armazenados pelos indivíduos podem ser manipulados e usados como forma de conexão e integração de informações que facilitam a compreensão de novos conhecimentos (MOREIRA, MASINI, 1982).

Ausubel (2000) elaborou sua Teoria da Aprendizagem Significativa ao verificar que abordagem tradicional era pouco produtora e que os conceitos adquiridos, na verdade, não eram interativos, por isso, não eram construídos e internalizados pelos alunos. Assim, a teoria proposta por ele, envolve uma análise das formas de ensinar e de aprender, examinando os contextos reais de sala de aula.

Conforme Ausubel (2000), a aprendizagem, para ser significativa, deve envolver os mecanismos, que devem possibilitar a mudança na maneira com que o aluno aprende, assim:

O conhecimento é significativo por definição. É o produto significativo de um processo psicológico cognitivo (“saber”) que envolve a interação entre ideias “logicamente” (culturalmente) significativas, ideias anteriores (“ancoradas”) relevantes da estrutura cognitiva particular do aprendiz (ou estrutura dos conhecimentos deste) e o “mecanismo”

mental do mesmo para aprender de forma significativa ou para adquirir e reter conhecimentos (AUSUBEL, 2000, p. 4).

Ao propor a aprendizagem significativa, Ausubel (2000), partiu do princípio de que os indivíduos só aprendem quando utilizam conteúdos existentes em sua estrutura cognitiva, chamados por ele de subsunçores, que servem como âncoras para interação entre os novos e conhecimentos anteriores. Para Moreira (2011, p.18) “subsunçor é, portanto, um conhecimento estabelecido na estrutura cognitiva do sujeito que aprende e que permite, por interação, dar significado a outros conhecimentos”.

Na perspectiva de Moreira (2011), a aprendizagem significativa é a caracterizada pela interação entre nossos conhecimentos existentes com os novos conhecimentos que vamos adquirir. O conhecimento pré-existente que será relevante a nova aprendizagem, pode ser representado de várias maneiras, como em forma de símbolo, um conceito, uma proposição, um modelo mental, entre outras formas. Já o pesquisador Rogers (2001) compreende a aprendizagem significativa como:

[...] uma aprendizagem que é mais do que uma acumulação de fatos. É uma aprendizagem que provoca uma modificação, quer seja no comportamento do indivíduo, na orientação futura que escolhe ou nas suas atitudes e personalidade. É uma aprendizagem penetrante, que não se limita a um aumento de conhecimento, mas que penetra profundamente todas as parcelas da sua existência (ROGER, 2001, p. 1).

De acordo com o entendimento de Rogers (2001), se o ensino e a aprendizagem não propiciam o desenvolvimento do aluno na perspectiva da aprendizagem significativa, é porque o que foi aprendido ocorreu de forma mecânica, literal, e sem reflexão, portanto, não contribuirá efetivamente para a construção do conhecimento do aluno. Neste contexto, para transformar uma aprendizagem mecânica em significativa. Há, portanto, necessidade que o discente produza os conhecimentos de forma a fazer relação com outros pré-existentes, para assumir uma função dinâmica, viva e participante, em um modelo capaz de estruturar e reestruturar as ideias antigas e novas, ordená-las por meio de suas interpretações.

Moreira e Mansani (1982) ressaltam a importância da escolha coerente dos subsunçores, para que sejam adequadamente utilizados. Conforme os autores os subsunçores devem ser relevantes na *ancoragem*¹ de novas ideias.

Novas idéias e informações podem ser aprendidas e retidas na medida em que conceitos relevantes e inclusivos estejam adequadamente claros e disponíveis na estrutura cognitiva do indivíduo e funcionem, dessa forma, como ponto de ancoragem para as novas ideias e conceitos (MOREIRA; MANSANI, 1982, p. 4).

Além da aprendizagem significativa por meio de conceitos, Ausubel (2000) apresenta a *Aprendizagem Proposicional* que consiste em proposições verbais com tarefas de aprendizagem, com o intuito de promover a interação de novos significados com ideias presentes na estrutura cognitiva dos indivíduos. Ausubel (2000, p.18) relata que, “neste caso, a tarefa de aprendizagem, ou proposição potencialmente significativa, consiste numa ideia compósita que se expressa verbalmente numa frase que contém significados de palavras quer denotativos, quer conotativos, e nas funções sintáticas”. Dessa forma, a aprendizagem significativa resulta de maneira própria, quando ocorre a relação dos conteúdos das proposições com os conteúdos de ideias relevantes preexistentes na estrutura cognitiva.

De acordo com Ausubel (2000), existem três formas de relações na *Aprendizagem Proposicional*, a subordinada (de subsunção), a subordinante e a combinatória. A subordinada é considerada a mais comum e simples na qual o novo conhecimento se “ancora” ao conhecimento preexistente adquirindo significado. A subordinante é aquela que promove a reorganização cognitiva conforme os conhecimentos relacionados e aprendidos por subsunção. Já a combinatória, trata da necessidade de um conjunto de conhecimentos prévios para que se construa um novo conhecimento, realizando a combinação entre as formas subordinada e subordinante. As formas de aprendizagem significativa, citadas acima, podem ser utilizadas como complementares, como uma sequência de ações que podem ser construídas para o processo de aprendizagem. Para Ausubel (2000, p. 18) a “maioria da aprendizagem proposicional é, obviamente, de subsunção ou combinatória”.

¹ Termo utilizado por Ausubel (2000) para definir a ligação entre os conhecimentos preexistentes na estrutura cognitiva do indivíduo, com novos conhecimentos.

Podemos dizer que a maioria das novas informações estão ligadas a conceitos ou proposições específicas e relevantes para a construção de um novo conhecimento, que envolve uma interação seletiva entre o novo e as ideias preexistentes na estrutura cognitiva, que Ausubel (2000) nomeou de ancoragem como maneira de sugerir a ligação com tais ideias construídas ao longo do tempo. “Por exemplo, no processo de subsunção, as ideias subordinantes preexistentes fornecem ancoragem à aprendizagem significativa de novas informações” (AUSUBEL, 2000, p.19).

Para que se desenvolva a aprendizagem significativa, são essencialmente consideradas duas condições. A primeira é a utilização de materiais potencialmente significativos, que possuam significados lógicos e que permitam ao aprendiz relacionar ideias-âncoras relevantes a partir do material. A segunda se configura na predisposição do aluno em querer aprender, pois se os indivíduos quiserem apenas memorizar os conteúdos de maneira literal, então a aprendizagem será mecânica e não significativa (PELIZZARE *et al.* 2002; MOREIRA, 2011). Ausubel (2000) ressalta que:

A aprendizagem significativa não é sinónimo de aprendizagem de material significativo. Em primeiro lugar, o material de aprendizagem apenas é *potencialmente* significativo. Em segundo, deve existir um mecanismo de aprendizagem significativa. O material de aprendizagem pode consistir em componentes já significativas (tais como pares de adjetivos), mas cada uma das componentes da tarefa da aprendizagem, bem como esta como um todo (aprender uma lista de palavras ligadas arbitrariamente), não são ‘logicamente’ significativas. Além disso, até mesmo o material logicamente significativo pode ser apreendido por memorização, caso o mecanismo de aprendizagem do aprendiz não seja significativo (AUSUBEL, 2000, p.17).

Ausubel (2000, p.20) distingue o processo de aprendizagem por memorização e o de aprendizagem significativa em que para ele “as tarefas de aprendizagem por memorização, como é óbvio, não se levam a cabo num vácuo cognitivo. Podem se relacionar com a estrutura cognitiva, mas *apenas* de uma forma arbitrária e literal que não resulta na aquisição de novos significados”. É o que muito acontece nos métodos tradicionais de ensino, no qual o processo de aprendizagem dos alunos se detém em aprender conteúdos, apenas com a teorização, sem a utilização de metodologias de aprendizagens voltadas às suas reais necessidades.

A capacidade de memorização, arbitrária e literal, ao relacionar tarefas com a estrutura cognitiva dos alunos promove consequências significativas para aprendizagem, porém Ausubel (2000, p. 20) coloca que nesse tipo de aprendizagem (arbitrária e literal) “apenas se conseguem interiorizar tarefas de aprendizagem relativamente simples e estas apenas conseguem ficar retidas por curtos períodos de tempo, a não ser que sejam bem apreendidas”, ou seja, o que os alunos aprendem facilmente é esquecido, pois não teve uma representatividade significativa relevante a sua estrutura cognitiva. Ausubel (2000) ainda reforça que:

É importante reconhecer-se que a aprendizagem significativa não implica que as novas informações formem um tipo de ligação simples com os elementos preexistentes na estrutura cognitiva. Pelo contrário, só na aprendizagem por memorização ocorre uma ligação simples, arbitrária e não integradora com a estrutura cognitiva preexistente. Na aprendizagem significativa, o mesmo processo de aquisição de informações resulta numa alteração quer das informações recentemente adquiridas, quer do aspecto especificamente relevante da estrutura cognitiva, à qual estão ligadas às novas informações (AUSUBEL, 2000, p. 19).

Podemos dizer que a aprendizagem só vale a pena quando o indivíduo consegue ampliar e reconfigurar aquilo que ele recebe. Devemos levar em consideração que os alunos já chegam com aprendizados prévios. Ao inserir a aprendizagem significativa no contexto de ensino, há a quebra do método tradicional que se configura em uma aprendizagem tradicional, com pouquíssimo ou sem nenhum significado.

As construções e reconstruções de conhecimentos a partir de situações vivenciadas pelos alunos, os motivam a aprender o novo conhecimento. “O ato de aprender deve ser, constantemente, um processo de reconstruções que permita diferentes tipos de relações entre fatos e objetos, tendo em vista a utilização dos saberes em diferentes situações. Ou seja, a aprendizagem deve ser significativa” (PINTO, 2012, p.78).

Moreira (2005), ainda ressalta que a aprendizagem significativa só acontece de fato, quando cinco elementos se encontram em harmonia no contexto escolar: o aluno, o professor, o conhecimento, o contexto e a avaliação. Quando esses elementos estão relacionados e integralizados, corroboram para que o aluno construa conhecimento.

Nesta perspectiva, pensamos em conceber a Metodologia 7Cs, com uma nova maneira de praticar os conteúdos básicos ministrados na disciplina de algoritmos,

assim como promover a aprendizagem significativa por meio da ancoragem dos conhecimentos, de uma fase para outra da metodologia. Além disso, buscamos estimular a troca de conhecimentos entre os alunos, para solucionar as atividades propostas com o uso da metodologia de maneira colaborativa. Segundo Moreira (2011), atividades colaborativas contribuem para o desenvolvimento da aprendizagem significativa, pois os alunos somam aos conhecimentos presentes em suas estruturas cognitivas novos conhecimentos.

Ao levar em consideração o exposto sobre aprendizagem significativa, é coerente propor o uso da Metodologia 7Cs, como meio para contribuir no processo de aprendizagem na disciplina de algoritmos, ao inserir elementos em sua concepção que considere uma nova forma de construção de conhecimentos com diferentes fases. Cada fase servirá de âncora para os alunos para obtenção de novos conhecimentos e assim oportunizar aprender significativamente.

2.2 APRENDIZAGEM SIGNIFICATIVA E O ENSINO DE ALGORITMOS

A aprendizagem significativa vem sendo alvo de alguns estudos voltados para a disciplina de algoritmos como estratégia de ensino em busca de minimizar as dificuldades de aprendizagem dos alunos. O autor Brezolin (2015) utilizou, em sua pesquisa, mapas conceituais na perspectiva da metodologia de aprendizagem significativa de Ausubel para a relação entre o conhecimento existente e o adquirido pelos alunos como forma de estruturação do conhecimento no processo de ensino e aprendizagem de algoritmos. Para ele “o planejamento do ensino, a utilização de técnicas apropriadas, a preparação do aluno, bem como a interação com o professor é fundamental nesse processo para que se obtenha uma aprendizagem significativa” (BREZOLIN, 2015, p.16)

Da mesma forma, Berssante (2016), em seu trabalho de dissertação, levanta a discussão sobre o uso da abordagem prática de algoritmos, baseando-se na teoria de aprendizagem significativa de Ausubel, que propõe minimizar o impacto das dificuldades enfrentadas pelos alunos e sair do método tradicional de ensino entrelaçado a teorização de forma mecânica e pouco significativa. De acordo com Brezolin (2015), a incorporação da aprendizagem significativa como metodologia de ensino de algoritmos pode viabilizar a construção de um ambiente de aprendizagem

que proporciona o dinamismo e adaptabilidade, permitindo que seja propício para o desenvolvimento autônomo do aluno em relação a construção do seu conhecimento.

Podemos dizer que aprendizagem na disciplina de algoritmos pode se tornar muito mais significativa no momento em que o professor busca novas estratégias ou metodologias de ensino e incorpora, em suas práticas, conhecimentos prévios dos alunos, possibilitando a apropriação dos conteúdos. Berssanette (2016) relata que é muito presente no ensino de algoritmos a ocorrência da aprendizagem mecânica. Para ele, esse fator se confirma quando os alunos sentem dificuldades em construir programas, mesmo que em sala demonstrem ter entendido o conteúdo trabalhado pelo professor. Outro aspecto a ser considerado é a forma de ensino do professor que apresenta os conteúdos da disciplina de algoritmos de forma isolada e a ausência de metodologias e/ou práticas de ensino adequadas (BERSSANETTE, 2016).

A necessidade de desenvolver uma proposta de ensino de algoritmos levou Barbosa (2011) a desenvolver uma metodologia inspirada na teoria de Ausubel, propondo uma maneira diferenciada de ensino em relação ao método tradicional empregado, somente com a teorização. O autor parte do princípio de que a aprendizagem é algo constante na estrutura cognitiva do aprendiz e todo e qualquer método pensando em facilitar e tornar o processo de aprendizagem mais esclarecedor deve ser utilizado para o enfrentamento das dificuldades em relação aos problemas de aprendizagem. Sua metodologia foi criada a partir de levantamentos de trabalhos relacionados às dificuldades inerentes a disciplina de algoritmos em que um dos maiores fatores relacionados a tais dificuldades se encontra na falta de motivação dos alunos.

Percebemos que os trabalhos desenvolvidos tem como foco a teoria de Aprendizagem Significativa de Ausubel em busca de solucionar os problemas vivenciados na disciplina de algoritmos, partindo da necessidade de criar novas metodologias que promovam motivação e minimizem dificuldades inerentes aos alunos. Todos os autores utilizaram como ferramenta de apoio a aprendizagem significativa, o mapa conceitual, na qual usaram como subsunçores conceitos relevantes a aprendizagem dos conteúdos da disciplina, abrindo a possibilidade de relação de conhecimentos preexistentes com o novo conhecimento trabalhado. Mas existem outras possibilidades e caminhos para o surgimento de outras propostas.

Na teoria da aprendizagem significativa criada por Ausubel (2000), os subsunçores estão relacionados a conceitos, que podem ser utilizados para despertar e auxiliar na construção de um novo conhecimento. Porém para Moreira (2011), devemos ver os subsunçores além de apenas conceitos, pois essa maneira de vê-los restringe a grandeza e possibilidades de uso e significado dos mesmos, ou seja, é melhor considerar o subsunçor como a escolha de um conhecimento relevante que buscamos para aprendizagem de um novo conhecimento. Dessa forma, a Metodologia 7Cs busca em sua própria estrutura, construir subsunçores, em cada fase, por meio do desenvolvimento de habilidades e competências, e construção de conhecimentos ancoras, para facilitar a compreensão dos conteúdos básicos de algoritmos.

3 CONCEPÇÃO DA METODOLOGIA 7CS

Neste capítulo apresentamos trabalhos sobre a criação de metodologias para disciplina de algoritmos, bem como o trabalho que inspirou para a concepção da Metodologia 7Cs. Relatamos, também, as observações participantes no primeiro e segundo semestre de 2018 na disciplina Algoritmos e Programação Estruturada de Computadores, uma pesquisa exploratória realizada no Instituto de Ciências Naturais e Exatas (ICEN) da Universidade Federal do Pará (UFPA) e, por fim, requisitos para concepção da metodologia.

3.1 METODOLOGIAS DESENVOLVIDAS PARA O ENSINO E APRENDIZAGEM DE ALGORITMOS

As problemáticas e dificuldades pertinentes aos processos de aprendizagem na disciplina de algoritmos, tem despertado estudos voltados a criação de novas metodologias ou uso de técnicas já existentes em busca de diminuir as dificuldades de compreensão sobre os conteúdos básicos. Tais dificuldades refletem não somente nos altos índices de retenção, mas também nos problemas presentes em disciplinas mais avançadas que possuem como pré-requisito uma boa base de algoritmos (PIMENTEL; OMAR, 2008).

Realizamos uma busca de trabalhos que apresentam a criação de metodologias voltadas à aprendizagem de algoritmos na qual destacamos até o momento duas dissertações e dois artigos. Para escolha dos trabalhos apresentados, foi levado em consideração os seguintes critérios: uso de metodologias diferenciadas para o ensino de algoritmos, aluno no centro do processo e resultados relevantes (Quadro 4).

Quadro 4 -Requisitos para pesquisa de metodologias

REQUISITOS DA PESQUISA	DESCRIÇÃO
Metodologias diferenciadas	Trabalhos que apresentam novas propostas e criações de metodologias para o ensino e aprendizagem de algoritmos.
Aluno no centro do processo	Trabalhos que promovem o envolvimento dos alunos, por meio da participação ativa nas atividades desenvolvidas e possibilitam a autonomia no processo de aprendizagem.
Resultados relevantes	Trabalhos que apresentam dados relevantes, na aplicação das metodologias desenvolvidas.

Fonte: Elaborado pelo autor.

Nesta perspectiva, a autora Petry (2005, p. 11) apresenta em seu trabalho de dissertação grandes contribuições acerca do ensino de algoritmos por meio de um “modelo computacional de um sistema de suporte ao ensino e aprendizagem no domínio de algoritmos”. Apesar de possuir elementos tradicionais, traz consigo a aprendizagem colaborativa a partir de um companheiro virtual de aprendizagem que utiliza a Modelagem Baseada em Restrições (*Constraint Based Modelling*) para representação do conhecimento e do raciocínio no desenvolvimento de algoritmos. A autora em seu estudo verificou que os problemas e dificuldades de aprendizagem na disciplina de algoritmos perpassam as limitações dos alunos, quando ressalta um ponto importante a ser considerado, a didática do professor em sala de aula. Afirma que a “didática utilizada pelo professor para apresentar o conteúdo introdutório da disciplina também é um fator fundamental no processo de ensino de algoritmos, pois para todo problema requer uma boa capacidade de abstração do aluno” (PETRY, 2005, p.3).

Como reafirmado neste trabalho, aprender algoritmos não é tarefa fácil para muitos, por isso a necessidade em incentivar e encorajar o uso de metodologias que dinamizem o ensino e a inclusão dos alunos na construção dos saberes. Sousa (2011) propôs a aplicação da metodologia ativa Aprendizagem Baseada em Problemas (PBL) para o ensino de conteúdos computacionais e algoritmos e técnicas de programação, no qual acredita que a estratégia metodológica da PBL rompe com os paradigmas do ensino tradicional e centraliza o conhecimento no aluno, valorizando o contexto e os seus conhecimentos prévios, potencializando o ensino e aprendizagem de computação em um curso de licenciatura.

Corroborar a essa ideia as palavras de (SOUSA, 2011, p.23), quando conclui que:

[...] parece ser fundamental encontrar uma estratégia de ensino e aprendizagem que seja capaz de desenvolver mecanismos para converter os conhecimentos prévios em aliados no processo de solução computacional de problemas para que o aluno seja capaz de perceber não somente o que ele conhece como também o que ele necessita conhecer, o que significa promover a autonomia da aprendizagem de tal forma que privilegie o metapensamento (SOUSA, 2011, p. 23).

No primeiro contato dos alunos com as atividades relacionadas à programação de computadores, geralmente são apresentados os aspectos sintáticos e semânticos das linguagens de programação (Pascal, C++, Python, entre outras), o que ocasiona a perda de alguns elementos fundamentais como o desenvolvimento do raciocínio lógico e a elaboração de hipóteses para solução de problemas (SOUSA, 2011).

Delgado *et al.* (2004) apresenta uma abordagem pedagógica para iniciar o estudo sobre algoritmos, como proposta para cursos superiores de computação, e propõe uma metodologia aplicada por um grupo de professores, como forma de concepção e implementação. “A metodologia se baseia na atuação do professor como facilitador, no papel proativo do aluno e em recursos didáticos voltados para o desenvolvimento da capacidade de abstração, do raciocínio lógico, da solução de problemas e da autonomia cognitiva (DELGADO *et al.*, 2004, p.4). O autor propôs três fases: a primeira trabalha com autonomia cognitiva, a segunda com a formalização das ideias e a última finalmente voltada a construção do algoritmo. Na metodologia o aluno é considerado o agente principal em que o “enfoque é o seu progresso nos aspectos de capacidade de abstração, do raciocínio lógico, da solução de problemas e da autonomia cognitiva” (DELGADO *et al.*, 2004, p.4).

Ao utilizar a metodologia ativa Aprendizagem Baseada em Problemas, Mourão (2017) relata que ministrar aulas relacionadas a algoritmos requer uma mudança de postura do professor em relação ao uso de métodos, técnicas, ferramentas, habilidades e dinâmicas inovadoras que possam somar ao processo de ensino e aprendizagem. A autora destaca que “a aprendizagem que envolve a motivação e novas formas de ensino, alcançando as dimensões afetivas e intelectuais, tendem a ser mais duradoura e sólida, aproximando professor e aluno em uma aprendizagem colaborativa” (MOURÃO, 2017, p.667). Por isso, é importante reforçar que o professor tem um papel fundamental em buscar e/ou criar novas possibilidades de abordagem do conteúdo de algoritmos, que

possam contribuir com a aprendizagem do aluno, levando em consideração os muitos desafios a serem enfrentados durante a disciplina.

Os autores Santos Fernandes e Freitas Junior (2016) realizaram uma revisão sistemática com o objetivo de analisar na literatura científica disponível trabalhos voltados ao desenvolvimentos de metodologias de ensino, criadas com o intuito de minimizar os fenômenos da reprovação e evasão em disciplinas voltadas a programação. A seguir no Quadro 5 apresentamos a sistematização realizada pelos autores, com atualização de novos trabalhos desenvolvidos e o acréscimo de estudos voltados ao uso de metodologias ativas para ensino de algoritmos.

Quadro 5 - Propostas metodológicas para o ensino de algoritmos (Continua...)

AUTORES	DESCRIÇÕES DAS PROPOSTAS METODOLOGICAS (DOS SANTOS FERNANDES E FREITAS JUNIOR, 2016)
Filho e Moura (2011) Rocha et al. (2010) Fassbinder, Paula, Araújo. (2012) Angelo et al. (2010) Gomes, Mendes (2012)	“Utilização do ambiente virtual de aprendizagem Moodle e seus recursos (chats, compartilhamento) para compartilhar materiais sem custo e respeitar o ritmo de cada estudante. Ao professor, cumpre o papel de acompanhar, aprimorar, treinar e gerenciar o aprendizado a partir da realização de encontros periódicos para avaliações individuais.”
Pereira, Medeiros, Menezes (2012) Inácio (2016)	“Utilização de software Scratch para exercitar a criatividade dos alunos através de um ambiente de fácil manuseio, tornando o aluno apto a solucionar problemas computacionais. Os professores devem utilizar esta ferramenta como auxílio para identificação do problema e o desenvolvimento da solução.”
Filho, Abegg, Simonetto. (2012)	“A lousa digital é uma ferramenta de auxílio utilizada para desenvolver estratégias no ensino da disciplina, propõe aos alunos discussão de conceitos em sala de aula. O professor tem a possibilidade de fazer anotações para melhorar o aprendizado dos alunos.”
Rapkiewicz et al (2006) Sales (2010) Salgado et al. (2013) Souza (2013)	“Metodologia baseada no ensino através de jogos, propondo o desenvolvimento de raciocínio lógico em fases introdutórias de programação. O jogo oferece construção de soluções através de representação lógica e apresenta ações concretas ao aluno.”
Silveira et al. (2015) Miranda, Pereira e Rissetti (2016) Lima e Leal (2010)	“A linguagem LOGO apresentada é uma ferramenta auxiliar ao “método tradicional” de aulas, ou seja, os alunos irão ter contato com a linguagem para organizar suas ideias e assim, solucionar os problemas propostos. Em relação a aula, utiliza-se o software conhecido como superLOGO 3.0”

Fonte: Adaptado Dos Santos Fernandes e Freitas Junior (2016, p. 3-4).

Quadro 5 - Propostas metodológicas para o ensino de algoritmos (conclusão)

AUTORES	DESCRIÇÕES DAS PROPOSTAS METODOLOGICAS (DOS SANTOS FERNANDES E FREITAS JUNIOR, 2016)
Fassbinder, Paula, Araújo (2012)	“A metodologia proposta é conhecida como competição de conhecimento capaz de oferecer aos alunos estímulos para provar o valor e conhecimento que obtiveram durante seus estudos. É considerada um estímulo para os alunos buscarem soluções. São criados grupos de alunos e organizados algumas olimpíadas de programação para que os alunos possam provar o conhecimento adquirido com a metodologia”
Galhardo, Zaina (2004)	“O modelo de ensino a partir de uma página web é um sistema, onde o professor disponibiliza os exercícios e permite ao aluno escolher os enunciados e através do sistema, cadastrar os atributos, as classes, etc para que no fim o sistema gere um código em java do programa feito. O ensino é realizado a distância e o professor pode acompanhar de qualquer lugar que tenha acesso a internet.”
Campos (2010)	“A ERM2C (Entender, revisar, melhorar, comparar e construir) capacita o discente para contribuir para uma melhoria após o “término” do algoritmo, ou seja, o aluno irá ser capacitado para criar soluções lógicas e o professor irá corrigir os algoritmos e ditar algumas correções, onde o aluno irá ser capaz de analisar e criar novas resoluções”
Caminha (2012)	“Nesta proposta metodológica, os estudantes recebem exemplos com conceitos utilizados na construção de jogos, diante disto, são feitas aulas práticas de melhorias do exemplo.”
Ambrósio (2010) Lima et al. (2016) Mourão (2017) Diemer (2019)	O uso de metodologias ativas no ensino de algoritmos, como o objetivo de tornar os alunos, protagonistas do seu processo de aprendizagem.

Fonte: Adaptado de Santos Fernandes e Freitas Junior (2016, p. 3-4).

De acordo com os autores Santos Fernandes e Freitas Junior (2016, p.1) “estas metodologias propõem, em sua maioria, a utilização de tecnologias que tornem dinâmico o aprendizado destas disciplinas, procurando considerar os limites dos estudantes e favorecer o seu próprio ritmo de estudos”. Poucas foram a metodologias de ensino criadas, como os trabalhos desenvolvidos por Delgado (2004) e Campos (2010), sem o uso de aparatos tecnológicos, para trabalhar a construção do conhecimento por meio da prática de algoritmos, por algoritmos.

Muitos foram os trabalhos desenvolvidos em busca de minimizar as dificuldades dos alunos em disciplinas voltadas a programação, que refletem um contexto problemático vivenciado, em relação aos altos índices de evasão em tais disciplinas (SANTOS FERNANDES; FREITAS JUNIOR, 2016). Mesmo com todas as propostas

existentes, esse contexto ainda continua ocorrendo, pois não existe uma solução concreta que possa dar conta de suprir todos os problemas relacionados as dificuldades dos alunos. Mas é possível por meio da criação dessas metodologias de ensino, minimizar e contribuir de forma significativa para compreensão dos conteúdos básicos de algoritmos.

Com base nos estudos apresentados e nas contribuições deixadas, destacaremos no Quadro 6 a seguir, alguns pontos que buscávamos e encontramos, os quais consideramos avanços e os pontos que sentimos falta em relação aos trabalhos desenvolvidos, as lacunas. Os pontos foram pensados a partir do nosso objetivo, com o que propomos neste trabalho de pesquisa.

Quadro 6 - Pontos levantados a partir das contribuições da literatura

AVANÇOS	LACUNAS
<ul style="list-style-type: none"> - Concepção de novas metodologias de ensino para disciplinas introdutórias a programação; - Professores em busca de soluções para minimizar as dificuldades dos alunos; - Protagonismo dos alunos em seu processo de aprendizagem; - Estimulo ao desenvolvimento de habilidades e competências, importantes na construção de algoritmos. 	<ul style="list-style-type: none"> - Poucos trabalhos desenvolvidos, direcionados ao estímulo de habilidades e competência, por meio da prática; - Poucas metodologias que estimulem o processo criativo dos alunos; - Poucas metodologias que promovam a ação de colaborar, para troca de conhecimentos entre os alunos.

Fonte: Elaborado pelo autor.

É interessante ressaltar que nos trabalhos apresentados até o momento na maior parte, relatam a importância de o professor incorporar em sua prática docente metodologias que proporcionem a aprendizagem dos alunos por meio de novas propostas de ensino. Por isso, desenvolvemos a Metodologia 7Cs o propósito de contribuir com professores e alunos, na busca de minimizar o recorrente cenário de reprovações e evasões em disciplinas voltadas a programação.

3.2 TRABALHO INSPIRADOR

Na busca de referências que pudessem embasar teoricamente uma nova estratégia de ensino na disciplina de algoritmos, encontramos o autor Campos (2009, 2010, 2011) que, ao ministrar a disciplina de lógica de programação, verificou que mais de 60% dos alunos haviam sido reprovados. Esse resultado o fez refletir se o problema estava somente relacionado a sua forma de ensino. Ao realizar um levantamento percebeu que muitas IES passam pelo mesmo dilema, o estimulando a repensar sua maneira de ensinar e buscar uma nova estratégia para ministrar a disciplina.

A partir desse contexto o autor criou a metodologia ERM²C (Entender, Revisar, Melhorar, Complementar e Construir) que visa contribuir com o processo de ensino-aprendizagem na disciplina de Lógica de Programação ou similar, buscando desenvolver capacidades para leitura de algoritmos prontos, identificar erros, propor modificações, inserir novas ações e por fim ter a capacidade pra construção de algoritmos.

Para o desenvolvimento da metodologia, o autor levou em consideração o objetivo da disciplina de lógica de programação, de garantir que ao final do processo que os alunos sejam capazes de desenvolver um algoritmo. Além disso, o autor também partiu da premissa de que antes de aprender a desenvolver algoritmos, primeiro é necessário ter a capacidade de ler/entender qualquer código construído, mesmo feito por outras pessoas.

A metodologia criada pelo autor segue uma sequência lógica de ações, trabalhando algoritmo por algoritmo. A metodologia de Campos foi validada e testada em turmas do ensino superior em cursos de Computação trazendo relevantes resultados para a disciplina de algoritmos e na aprendizagem dos alunos. Partindo desse potencial o trabalho desenvolvido pelo autor serviu como inspiração para criação da Metodologia 7Cs, em desenvolver uma proposta também para trabalhar algoritmos por algoritmos, com o encadeamento lógico para construção do conhecimento.

3.3 OBSERVAÇÃO PARTICIPANTE

Na busca de propor uma nova maneira de aprendizagem na disciplina de algoritmos, partimos da premissa que é importante vivenciar e perceber o ambiente e

o público ao qual destinamos nosso produto. Dessa maneira, realizamos a observação participante em dois cursos de graduação da UFPA, que nos proporcionou grandes contribuições para concepção do projeto de pesquisa a ser desenvolvido.

A observação participante é um método em que o pesquisador se insere no meio e com o grupo a qual estará investigando, possibilitando uma maior profundidade e conhecimentos que o levarão para as interpretações e interações entre o pesquisador e o grupo pesquisado. “Esse tipo de investigação, fundamentado em descobertas no campo, envolve a participação do pesquisador no dia a dia dos pesquisados. Por isso, a observação participante deve ser personalizada e multifatorial, requerendo um compromisso de longo prazo” (ABIB; HOPPEN; HAYASHI JÚNIOR, 2013, p. 605). Na observação participante é importante que o pesquisador se torne parte do ambiente, pois abrirá a possibilidade de melhor entendimento e compreensão do contexto e tudo o que envolve o mesmo.

A observação participante, ou observação ativa, consiste na participação real do conhecimento na vida da comunidade, do grupo ou de uma situação determinada. Neste caso, o observador assume, pelo menos até certo ponto, o papel de um membro do grupo. Daí por que se pode definir observação participante como a técnica pela qual se chega ao conhecimento da vida de um grupo a partir do interior dele mesmo (GIL, 2008, p. 103).

A observação participante é um método muito utilizado e que abre grandes possibilidades de sucesso para o pesquisador, principalmente em relação a profundidade e vivência que ela proporciona durante seu processo e que envolve diversos fatores. De acordo com Proença (2007) a relevância e aplicabilidade desse método consiste em alguns fatores como tempo e lugar:

- a) **Tempo:** quanto mais for o contato do observador com o ambiente mais possibilidades de interpretação serão alcançadas e estarão mais próximas a realidade, podendo promover uma relação de intimidade e confiabilidade entre o pesquisador e os envolvidos no processo.
- b) **Lugar:** O pesquisador deve levar em consideração que as condições físicas do ambiente observado também influenciam o processo investigado. É necessário que para além dos registros observados, registrar o ambiente físico onde ocorre todas as atividades.

É importante para os pesquisadores que pretendem ir a campo com intenção de utilizar o método de observação participante leve em consideração esses fatores, pois farão grande diferença em seu resultado e interpretação dos dados coletados no final da observação. Além dos fatores citados, Gil (2008) aponta duas maneiras distintas de observação participantes que são:

- a) **Natural:** nesse processo o observador deve pertencer à mesma comunidade ou ao grupo que está investigando.
- b) **Artificial:** o observador se integra ao grupo com o objetivo de realizar uma investigação.

O autor ainda ressalta que na observação artificial, o observador tende a se deparar com mais problemas do que na observação natural. Levando em consideração as duas maneiras distintas e no que foi feito para presente pesquisa até o momento, acreditamos que foi realizada e a observação participante artificial. A seguir serão descritas as observações realizadas no primeiro e segundo semestre de 2018 em duas turmas de diferentes áreas, Ciências da Computação e Engenharia Elétrica, que possuem em comum a disciplina de algoritmos.

3.3.1 Relato da primeira observação participante

A primeira observação consistiu no conhecimento da rotina, na compreensão das dificuldades dos discentes na disciplina de algoritmos, dos conteúdos ministrados em sala de aula e, na busca das possíveis soluções dos problemas que seriam apresentados durante o processo na disciplina. Realizamos a observação participante no primeiro semestre de 2018, no curso de Bacharelado em Ciência da Computação, no Instituto de Ciências Exatas e Naturais (ICEN). Estavam presentes em sala de aula, a professora da disciplina, um monitor e um mestrando do PPGCIMES. A participação em sala de aula se dava no auxílio aos discentes, na resolução das atividades. Esse processo nos possibilitou maior proximidade entre o pesquisador e grupo observado.

No processo de Observação Participante é importante levar em consideração o ambiente ao qual o pesquisador está observando e como pode estar influenciando no

processo, nesse caso, na construção do conhecimento na disciplina de algoritmos. Dependendo do contexto em relação ao ambiente e a rotina dos discentes em sala, pode ser alterado positiva ou negativamente.

Com a observação constatamos que os alunos do curso de graduação Ciências da Computação dispõem de um laboratório de informática, muito bem estruturado, contando com as condições necessárias para praticar e realizar as atividades durante as aulas. Esse é um aspecto muito positivo e que poderia interferir no processo de aprendizagem, caso fosse contrário. Mas não se trata apenas do ambiente, as dificuldades dos alunos na disciplina são em decorrências de vários fatores, entre eles: a exigência de habilidades, que não estão relacionadas diretamente com ambientação e sim com a base teórica trazida do ensino básico.

Identificamos que no início das aulas houve uma grande participação dos alunos, principalmente pelo fato de estarem em um processo de descoberta tanto da disciplina como no contexto da vida acadêmica. A disciplina é ministrada geralmente no primeiro semestre, e no primeiro ano de curso. O fato de serem calouros pode ser um fator que promove essa participação e alta motivação inicial, assim como as estratégias utilizadas pela professora. Porém, após a primeira avaliação foi nítida a decepção de alguns pela nota, causou um sentimento de insatisfação consigo mesmo e desmotivação.

A construção de algoritmos é algo preciso e que exige que o aluno realize um código sem erros para que possa executá-lo. Na correção das provas os discentes tinham a oportunidade de contestar suas notas se necessário e se não, tinham a possibilidade de identificar seus erros. Alguns erros eram mínimos, mas que dentro da estrutura de um algoritmo, fazem muita diferença. Os alunos precisam ser constantemente motivados para que ao continuar o andamento da disciplina continuem participando ativamente das aulas e na busca de compreender os conteúdos seguintes.

São vários fatores que influenciam e refletem nas notas obtidas pela turma. Sabemos que além da disciplina de algoritmos os discentes cursam outras, esse é um fator que mudava também a rotina em sala de aula. Em dia de prova de outras disciplinas, os alunos não se concentravam tanto o quanto necessário para aprender o que lhes eram ensinados. Concluímos que o comportamento dos discentes em sala de aula se torna instável e que a rotina dependia muito de quão estavam motivados no momento.

Para observação, criamos um instrumento (Apêndice I) para analisar de maneira mais específica alguns pontos para o trabalho de dissertação. O instrumento utilizado foi um roteiro de observação com questões relevantes. Serão destacadas algumas questões presentes no roteiro e pontos gerais identificados durante a disciplina de algoritmos.

Com o intuito de verificar qual a forma mais adequada ou que mais despertava o interesse dos discentes para resolução das atividades em sala de aula, buscamos responder a seguinte questão *“As atividades propostas para a turma foram de natureza individual ou coletiva?”*. Mesmo em atividades individuais era perceptível na turma que a colaboração era uma competência inerente a eles, porém não permitia discussões mais profundas sobre as questões e as soluções dos problemas. Já nas atividades coletivas os alunos tinham a oportunidade de discutir e realizar trocar conhecimentos que contribuem para agregação de novas estratégias de solução de problemas. Um ponto importante a ser considerado é que nas atividades coletivas eram desenvolvidas algumas competências como a interatividade, aprendizagem colaborativa, a descoberta de novas possibilidades de solução de problemas.

Ao considerar como uma disciplina complexa, a pergunta *“As atividades e os problemas propostos são desafiadores e proveitosos para a aprendizagem dos discentes?”* tinha o objetivo de verificar se as atividades e desafios propostos se construíram para contribuir de maneira significativa para os alunos. Todas atividades relacionadas a concepção de algoritmos se tornavam muito valiosas para construção do conhecimento dos alunos acerca da disciplina, pois a partir das atividades eles podiam identificar quais a suas limitações e em que poderiam melhorar. Em algumas situações, a professora da disciplina realizava alguns desafios individualmente em que os discentes ao conseguirem responder e apresentar a solução do problema em um determinado tempo, era premiado com um ponto extra para sua nota final. Os discentes nessa situação ficaram muito motivados, o que aumentou o interesse sobre a disciplina e pelas atividades propostas.

Durante todo o processo verificamos a relação entre a professora e monitores da disciplina e os alunos, com base na pergunta *“Existe uma boa relação entre os alunos, professora e monitores?”*. A professora da disciplina e seus monitores estiveram atentos aos discentes e disponíveis para auxiliar no que fosse necessário para resolução das atividades. Um fator interessante é que mesmo com uma boa relação aluno-professor,

alguns discentes não solicitavam ajuda, principalmente em relação a compreensão do conteúdo e de algumas especificidades de estruturas de algoritmos.

Após a explicação de um conteúdo, por exemplo, poucos ou nenhum dos alunos questionavam ou aproveitavam o momento para tirar suas dúvidas sobre os conhecimentos abordados, mesmo a professora perguntando se alguém tinha alguma dúvida a ser esclarecida. Dessa maneira, acreditamos que os alunos haviam entendido o conteúdo e não teriam problemas para colocarem em prática. Mas os resultados das provas mostravam o contrário.

Na busca de compreender e verificar quais possibilidades para facilitar o ensino de algoritmo e programação, foi importante identificar "*Quais as principais dificuldades dos alunos durante a aula e atividades propostas?*". Muitas são as dificuldades dos alunos em relação a disciplina, mas destacamos as dificuldades mais recorrentes entre os alunos que foram:

Interpretação do comando: muitos não conseguiam assimilar o que era necessário de um comando, interpretando de maneira errada. Esse fato era visível quando ao realizar a construção de um algoritmo os alunos não conseguiam contemplar todos os elementos que constavam no comando, resultando em uma solução ineficiente.

Conceitos matemáticos: possuíam dificuldades em compreender certas regras de matemáticas, como a prioridade dos operadores.

Organização das ideias (pensamento lógico): ao ler o comando dos problemas foi perceptível a dificuldade em organizar dentro dos programas uma linha de pensamento lógico condizente e ordenada para solução.

A observação realizada na turma foi importante para que pudéssemos conhecer e reconhecer de fato o contexto dos alunos na disciplina, a rotina dos alunos, fatores motivacionais, as dificuldades e descoberta de novos elementos a serem considerados para a presente pesquisa.

3.3.2 Relato da segunda observação

A segunda observação participante ocorreu durante o estágio supervisionado em uma turma de graduação no curso de Engenharia Elétrica no segundo semestre de 2018. As aulas ocorreram no Instituto de Ciências Exatas e Naturais (ICEN). Estavam

presentes em sala de aula a professora da disciplina e um monitor. Foi a partir dessa experiência que surgiram novas possibilidades para construção do produto para o PPGCIMES. Destacaremos alguns pontos específicos a partir da observação.

A escolha do estágio na disciplina de Programação Estruturada de Computadores foi realizada com o propósito de realizar testes de atividades e dinâmicas em sala de aula que pudessem auxiliar a professora e os alunos, na busca de facilitar a compreensão da disciplina com uma proposta diferente. Ao contrário da primeira observação, na qual seguíamos um roteiro para observar situações mais específicas, queríamos estar abertos a outras dinâmicas e fenômenos que ocorriam durante as aulas.

Durante a observação, foi possível identificar alguns aspectos que não havíamos percebido anteriormente. Verificamos que todos os alunos estavam muito envolvidos e que a cada início de conteúdo pareciam bem motivados. No decorrer das atividades o grau de complexidade aumentava e as dificuldades em compreender o que estavam aprendendo aumentavam. As dificuldades se evidenciaram no momento da realização das provas, na qual inicialmente tiveram médias não satisfatórias.

Apesar de em sala demonstrarem ter compreendido com o que a professora havia ensinado, isso não se refletia nos resultados das provas. Na primeira prova, algo que nos chamou atenção foi os alunos errarem questões que exigiam deles interpretar e compreender trechos de códigos simples. Nesse ponto percebemos o quanto a compreensão e interpretação são dificuldades presentes e relevantes a serem mais trabalhadas e desenvolvidas na disciplina.

É incrível como os resultados das notas afetam os alunos. Assim como na primeira observação, identificamos que muitos se sentiram tristes e desmotivados, pois de certa forma sabiam que não seria uma tarefa fácil dali para frente. A professora ao entregar as provas corrigia no quadro e os alunos percebiam que em algumas questões continham erros mínimos, mas que dentro da estrutura de um algoritmo fazem toda diferença.

Nas aulas os alunos utilizavam o computador para realizar as atividades. O exercício na linguagem de programação permite a tentativa e o erro. Então se o aluno erra o código ao executar o programa irá indicar que algo está errado, tendo a oportunidade de corrigir e melhorar o código. Nas provas as questões são feitas à mão e não possuem a facilidade de testar e verificar se que escreveu está de acordo com o

que o programa realizou. Por esse motivo sempre os incentivamos a realizarem as atividades também de maneira escrita, mas percebemos que poucos realizavam esse exercício. Esse também pode ter contribuído para as dificuldades encontradas nas provas.

Outro aspecto observado está relacionado a alunos que chegavam depois do horário de início da aula, quando a professora havia explicado a parte conceitual da estrutura a qual estava trabalhando. Esses alunos ficavam desorientados e tinham dificuldades em acompanhar os demais. Cada minuto de aula na disciplina de algoritmos é importantíssimo, pois segue uma sequência de conteúdos que se complementam. Nesse sentido a contribuição, na partilha de informações sobre os conteúdos e atividades se fez muito presente entre os discentes na turma. A forma com um aluno explicava as maneiras de resolução das atividades para o colega do lado, tinha uma particularidade interessante, que facilita a compreensão e o esclarecimento de dúvidas individuais.

Na disciplina os alunos tinham listas de exercícios específicas para realizar em sala e outra para realizarem em casa, para que pudessem praticar e enviar em determinado prazo para pontuação extra. Constatamos que os alunos que apresentaram menor dificuldade nas avaliações, eram os que se esforçavam para entregar todas as listas, já os que precisavam de pontuação, não conseguiam entregar todas. Em uma conversa informal com um dos alunos da turma realizamos a seguinte pergunta: “Você resolveu as listas de exercícios? Você pratica em casa?”, o aluno respondeu que possui muitas dificuldades e que além da disciplina de algoritmos estava se dedicando a outras disciplinas do curso que as considera complexas, e que não pratica a resolução de problemas em casa somente em sala. A disciplina exige que os alunos aprimorem seus conhecimentos e prática são muito importantes para o processo de construção do conhecimento sobre a aprendizagem de algoritmos.

Foi durante as observações no estágio que tivemos a ideia de realizar a concepção de uma metodologia de aprendizagem para auxiliar os alunos, propondo uma nova maneira de construção do conhecimento a partir de uma sequência lógica de ações para a prática da disciplina de algoritmos que pudesse estimular o desenvolvimento de habilidades e competências, nas quais os alunos possuíam dificuldades. Ambas as observações serviram para que pudessemos verificar os

aspectos positivos e negativos no desenvolvimento das aulas, retirando requisitos para a concepção da Metodologia 7Cs.

3.4 RELATO DE PESQUISA EXPLORATÓRIA

Na disciplina de Planejamento e Produção de Materiais Educacionais Multimidiáticos, ministrada no segundo semestre de 2018, no PPGCIMES, foi realizada com a turma uma atividade que consistiu em realizar uma conversa com o público para os quais os mestrandos estão realizando seus projetos. Essa atividade realizamos com alunos do curso de Engenharia Elétrica da UFPA que estavam cursando a disciplina de algoritmos e na qual estávamos no estágio supervisionado. A proposta nos possibilitou a identificação de aspectos importantes para o andamento da pesquisa e da proposta do processo e produto pretendido para dissertação. O desenvolvimento da dinâmica foi realizado da seguinte maneira:

- Explicação da atividade, seu objetivo e como a dinâmica iria acontecer.
- Os alunos foram divididos em cinco grupos.
- Cada grupo ficou com uma pergunta sorteada.
- Nos grupos os integrantes tiveram as seguintes funções:
 - a) Todos deveriam discutir e expor sua opinião sobre a pergunta;
 - b) Um aluno ficaria responsável por colocar no papel a opinião de todos do grupo;
 - c) A apresentação do trabalho foi realizada por um representante de cada grupo.

Os alunos tiveram dez minutos para discutir e colocar no papel a resposta contemplando a opinião de todos. O objetivo era que todos os alunos pudessem participar e expor suas opiniões de alguma maneira sobre as perguntas. As perguntas foram pensadas para que pudéssemos extrair informações importantes, e ter a percepção deles sobre algumas premissas em relação à disciplina. Para preservar a identidade dos participantes, os identificamos como A1, A2 e A3. Segue as pergunta e breve considerações sobre as respostas:

1. Antes da graduação vocês tiveram muito, pouco ou nenhum conhecimento sobre algoritmo?

O grupo era formado por quatro integrantes e deles dois já haviam tido contato com algoritmos, porém pouco lembravam. Abrimos a pergunta para turma e, de 23 alunos, apenas cinco tiveram alguma experiência prévia sobre algoritmos. A maioria da turma estava passando por um período de descoberta da disciplina. Essa é uma realidade de muitos cursos da computação como discutido anteriormente, pois muitos deles haviam saído do ensino médio para graduação. Um aspecto interessante e relatado pelos alunos, é que nem mesmo sabiam que no curso de Engenharia Elétrica teria essa disciplina.

2. Quais habilidades vocês acreditam ser necessárias para um bom desempenho na disciplina?

O grupo era formado por cinco integrantes e listaram cinco habilidades que consideram importantes para construção de algoritmos e para um bom desempenho na disciplina. As habilidades listadas e resposta de acordo com o aluno A1:

- a) “Raciocínio Lógico, tem que entender de lógica para poder desenvolver códigos”;
- b) “Noções Aritméticas, por que computação é basicamente matemática;”
- c) “Ser bom em interpretar problemas;”
- d) “Conhecer a linguagem que está se programando;”
- e) “Praticar.”

As habilidades listadas a partir da percepção dos alunos, se concretizam com as listadas na literatura sobre a disciplina. O interessante é que ainda estavam no início do semestre e haviam tido apenas uma aula prática. Foi importante perceber que os alunos já sentem desde o início da disciplina a exigência de habilidades inerentes a construção de um algoritmo.

- 3) Com base nas atividades já realizadas e dificuldades possivelmente encontradas, quais habilidades precisam ser melhoradas?

O grupo era formado por quatro integrantes. A resposta dada pelo aluno A2 foi:

A2: “desenvolver a habilidade de compreensão e abstração dos comandos dentro do algoritmo, a lógica matemática que não fica tão claro”.

A2: “Em relação diretamente a programação uma das principais dificuldades é gravar os códigos do algoritmo em si, saber o que cada coisinha serve, e de que forma vai ser usado.”

Como já citado, os alunos apenas haviam passado por uma aula prática, o que possibilitou a resposta. No grupo foi discutido que cada um possuía uma dificuldade, mas que acabam se convergindo, pois uma dependia da outra. A lógica matemática se faz muito presente na disciplina, pois muitas das questões desenvolvidas envolvem conceitos matemáticos.

4) Quais as dificuldades que vocês acreditam que irão ter ou já tiveram nos primeiros programas realizados na disciplina?

Os alunos conseguiram identificar cinco dificuldades nas primeiras atividades.

Resposta aluno A3:

- a) “Sintaxe”
- b) “Ordem dos comandos”
- c) “Abstração em relação a dificuldade de transcrever a ideia do comando em si”
- d) “A troca de linguagem”
- e) “Trabalho em equipe por várias pessoas pensarem de uma maneira diferente, embate de ideias.”

Relataram que ainda é muito confuso os comandos e onde podem ou não os colocar e que também sentem dificuldades em interpretar os comandos de maneira a transcrever a ideia de forma coerente. Ao abrir para turma acrescentaram que além das dificuldades citadas, as próprias habilidades que são necessárias respondida na segunda pergunta, estão implícitas nas dificuldades expostas pelos colegas.

5) Quais suas expectativas em relação a disciplina?

São calouros e criaram grandes expectativas, em suas falas relataram que pensavam que iam logo virar hackers e outros nem imaginavam que iam estudar essa disciplina no curso e alguns falaram que nem mesmo sabia que existia programação. A expectativa é que no final da disciplina e do semestre consigam programar de maneira eficiente e passar com excelente em todos nas provas. Isso mostra que os alunos criam boas expectativas sobre a disciplina, e o que os fazem evadir e muitas vezes desistir é a frustração por não alcançar a meta e expectativa desejada, pois no decorrer da disciplina se deparam com uma realidade totalmente diferente do que imaginavam.

A dinâmica contou com a participação de 23 alunos do curso de Engenharia Elétrica, e todos puderam participar. A atividade fez com algumas premissas fossem confirmadas e outras desconstruídas. Foi importante perceber a visão dos alunos, fazendo reflexões sobre suas próprias necessidades.

3.5 REQUISITOS PARA CONCEPÇÃO DA METODOLOGIA 7CS

De acordo com objetivo desta pesquisa, elencamos alguns requisitos básicos necessários a concepção da Metodologia 7Cs. A seguir no Quadro 7 apresentamos os requisitos e suas respectivas descrições.

Quadro 7 - Requisitos para concepção da Metodologia 7Cs

REQUISITOS	DESCRIÇÃO
Ter caráter criativo e inovador	A Metodologia 7Cs deve ser concebida considerando princípios criativos e inovadores, pensados para o contexto a qual será desenvolvida.
Promover a aprendizagem significativa	A Metodologia 7Cs deve possibilitar a aprendizagem significativa, como fator motivacional para processo de aprendizagem sobre a disciplina de algoritmos.
Facilitar a compreensão dos conteúdos	A Metodologia 7Cs deve propor uma nova maneira de construção de conhecimento que facilite a compreensão dos conteúdos básicos de algoritmos, buscando desenvolver conhecimentos prévios.
Estimular habilidades e competências	A Metodologia 7Cs deve estimular habilidades e competências exigidas na disciplina de algoritmos, tais como a capacidade de interpretar os comandos, de solucionar problemas, de abstração, entre outras.
Realizar o <i>feedback</i> em tempo hábil	A Metodologia 7Cs deve em tempo hábil alcançar minimamente as lacunas existentes, em relação as dificuldades de compreensão dos alunos sobre os conteúdos da disciplina algoritmos.

Fonte: Elaborado pelo autor.

Os requisitos foram escolhidos a partir das pesquisas bibliográficas realizadas sobre as dificuldades vivenciadas pelos estudantes na compreensão dos conteúdos de algoritmos, das observações participantes relatadas anteriormente nas disciplinas de algoritmos da graduação, de critérios relacionados a criatividade e inovação e pautada na teoria da aprendizagem significativa de Ausubel (2000).

4 CARACTERÍSTICAS DA METODOLOGIA 7Cs

Neste capítulo abordamos a descrição da Metodologia 7Cs para apresentar cada uma das dimensões, seus objetivos, as habilidades e competências que buscamos desenvolver, bem como os pré-requisitos necessários. Descreveremos, também, as fases da metodologia, ao considerar a Teoria de Aprendizagem Significativa de Ausubel (2000). Além disso, classificaremos a categoria da metodologia como produto educacional para esta proposta de pesquisa e exemplos de aplicação em forma de aulas.

4.1 DESCRIÇÃO DA METODOLOGIA 7CS

Como proposta deste Mestrado Profissional concebemos, desenvolvemos e testamos a Metodologia 7Cs, de acordo com a proposta de produtos educacionais, está orientada à “conjugação de ações em sequências orientadas para um fim comum, com objetivo de criar, inventar, projetar, transformar, produzir, controlar, manter e usar produtos e sistemas de ensino-aprendizagem” (MALCHER; ELIASQUEVICI; VELOSO, 2018).

Dessa maneira, podemos dizer que a Metodologia 7Cs se enquadra como processo educacional, das características citadas, como uma metodologia criativa e inovadora que busca transformar o modo de estruturação de aprendizagem na sistematização da prática em disciplinas introdutórias a programação, em busca de proporcionar a aprendizagem significativa dos alunos e auxílio ao professor em seu processo de ensino.

A partir da premissa de que a maioria dos alunos possuem pouca ou nenhuma experiência com a construção de algoritmos (CAMPOS, 2011) e que possuem insuficiências em habilidades e competências, como a interpretação textual, a capacidade de abstração, a solução de problemas, conceitos matemáticos, entre outras (BERSSANETTE, 2016), exigidas pela disciplina, pensamos que dificilmente os alunos conseguirão aprender a construir algoritmos sem dificuldades e com eficiência, se não conseguirem transpor as dificuldades existentes.

Nesse sentido a Metodologia 7Cs, é uma metodologia de ensino e aprendizagem, que possui como proposta facilitar a compreensão dos conteúdos básicos de disciplinas introdutórias a programação, com o intuito de desenvolver habilidades e competências,

a partir do desenvolvimento de uma sequência lógica de para o encadeamento dos conhecimentos, na estrutura cognitiva dos alunos.

A metodologia contém em sua estrutura sete dimensões (Compreender, Conceber, Completar, Compatibilizar, Corrigir, Construir e Criar), cada uma delas possui um objetivo, que traça um caminho para construção do conhecimento sobre os conteúdos trabalhados na disciplina. O Quadro 8, apresenta a descrição de cada uma das dimensões propostas, com seus objetivos e habilidades e competências que se pretende que os alunos desenvolvam e os pré-requisitos. Além das dimensões da metodologia a ação **Colaborar**, está prevista de maneira transversal, para estimular em todas as dimensões atividades colaborativas. Segundo Moreira (2011, p.50), a colaboração potencializa a aprendizagem significativa, por viabilizar a troca de informações e a negociação dos significados.

Quadro 8 - Descrição da Metodologia 7Cs (continua...)

	DIMENSÃO	DESCRIÇÃO	OBJETIVOS	HABILIDADES E COMPETÊNCIAS	PRÉ-REQUISITOS
C O L A B O R A R	Compreender	Os alunos devem compreender um novo conhecimento, por meio da interpretação e análise de algoritmos simples.	<ul style="list-style-type: none"> Estimular a capacidade de interpretar trechos de algoritmos ou algoritmos inteiros simples, já construídos. 	<ul style="list-style-type: none"> Compreender e interpretar os funcionamentos da estrutura de um algoritmo. Desenvolver a capacidade de solução de problemas. 	<ul style="list-style-type: none"> Conhecer os conceitos básicos da estrutura a ser trabalhada.
	Conceber	Após passar pelo processo de compreensão, os alunos concebem seus primeiros algoritmos, a partir de situações problemas dadas.	<ul style="list-style-type: none"> Estimular a capacidade de solucionar problemas, ao conceber seus primeiros algoritmos. 	<ul style="list-style-type: none"> Interpretar os comandos das questões. Desenvolver a capacidade de solução de problemas. 	<ul style="list-style-type: none"> Passar pela dimensão Compreender. Conhecer minimamente a linguagem de programação a ser utilizada pelo professor.
	Completar	Partindo do princípio de que os alunos compreenderam o conteúdo e que foram capazes de conceber seus primeiros algoritmos, serão capazes de analisar um algoritmo incompleto e perceber e completar o está faltando para ser executado.	<ul style="list-style-type: none"> Estimular a capacidade de analisar um algoritmo já construído, para identificar os elementos que faltam para se tornar executável. 	<ul style="list-style-type: none"> Interpretar os comandos das questões e códigos. Desenvolver a capacidade de percepção de erros e de ajuste de algoritmo. 	<ul style="list-style-type: none"> Passar pelas dimensões Compreender e Conceber. Saber os elementos necessários para execução de um algoritmo.
	Compatibilizar	Os alunos devem, a partir de um algoritmo apresentado com a sequência de passos de forma desordenada, identificar quais comandos devem ser realocados e compatibilizá-los, para que possa ser executável.	<ul style="list-style-type: none"> Estimular a capacidade de organização lógica e encadeamento dos passos na construção de um algoritmo. 	<ul style="list-style-type: none"> Interpretar os comandos das questões. Desenvolver a capacidade de organizar o encadeamento das ideias. Desenvolver a capacidade de ajuste de algoritmo. 	<ul style="list-style-type: none"> Passar pelas dimensões Compreender, Conceber e Completar. Saber os elementos necessários para execução de um algoritmo.

Fonte: Elaborado pelo autor.

Quadro 8 - **Descrição** da Metodologia 7Cs (Conclusão)

	DIMENSÃO	DESCRIÇÃO	OBJETIVOS	HABILIDADES E COMPETÊNCIAS	PRÉ-REQUISITOS
C O L O R A R	Corrigir	Os alunos terão que analisar e corrigir algoritmos implementados com erros diversificados.	<ul style="list-style-type: none"> Estimular a capacidade de analisar e perceber erros em algoritmos já construídos. 	<ul style="list-style-type: none"> Corrigir códigos com erros. Desenvolver a capacidade de análise de algoritmos. Desenvolver a capacidade de ajuste de algoritmo. 	<ul style="list-style-type: none"> Passar pelas dimensões Compreender, Conceber, Completar e Compatibilizar. Saber os elementos necessários para execução de um algoritmo.
	Construir	Os alunos devem utilizar todos os conhecimentos já adquiridos, para construir a solução de problemas mais complexos.	<ul style="list-style-type: none"> Estimular a solução de problemas que necessitam da construção de algoritmos com nível de complexidade maior. 	<ul style="list-style-type: none"> Interpretar os comandos das questões. Desenvolver a capacidade de solução de problema. Construir algoritmos com eficiência. 	<ul style="list-style-type: none"> Passar pelas dimensões Compreender, Conceber, Completar e Compatibilizar e Corrigir. Saber interpretar os comandos, analisar, organizar, corrigir e solucionar.
	Criar	Os alunos deverão criar uma situação problema, em forma de comando de questão, com certo nível de complexidade, mas viável de ser solucionado.	<ul style="list-style-type: none"> Estimular a solução de problemas e o a capacidade de criar, a partir dos conhecimentos adquiridos anteriormente. 	<ul style="list-style-type: none"> A capacidade de trabalhar em grupo. Aprender de maneira colaborativa. Potencializar a solução de problemas. Estimular o processo criativo. 	<ul style="list-style-type: none"> Passar pelas dimensões Compreender, Conceber, Completar e Compatibilizar e Corrigir e Construir; Ter conhecimentos para criar um comando de questão, interpretar comandos, e saber solucionar problemas.

Fonte: Elaborado pelo autor.

É válido ressaltar que a metodologia foi desenvolvida para trabalhar a parte prática de algoritmos, portanto, é essencial a explicação pelo professor, sobre cada estrutura de algoritmos (Estrutura Sequencial, Estrutura de Seleção e Estrutura de Repetição, entre outras), antes de iniciar o seu uso. A parte teórica também é muito importante, pois a compreensão do conteúdo pode influenciar diretamente no desempenho dos alunos em solucionar problemas e iniciar o uso da Metodologia 7Cs. A metodologia busca potencializar tal compreensão.

Como já relatado, para Ausubel (2000), os indivíduos só aprendem significativamente quando utilizam conteúdos existentes em sua estrutura cognitiva, chamados por ele de subsunçores, que servem como âncoras para interação entre os novos e os conhecimentos anteriores. A teoria de Ausubel (2000) subdivide-se em três categorias (Subordinada, Superordenada e Combinatória), apresentadas no Quadro 9 a seguir.

Quadro 9 - Categorias de aprendizagem

CATEGORIAS	DESCRIÇÃO
Subordinada	Na aprendizagem subordinada o novo conhecimento passa a interagir com os subsunçores (conhecimentos preexistentes do indivíduo) para tornar o novo conhecimento cheio de significados.
Superordenada	Na aprendizagem superordenada a partir dos subsunçores adquiridos o indivíduo, consegue concretizar uma ideia mais geral e organizar os conhecimentos preexistentes como parte dessa ideia.
Combinatória	Na aprendizagem Combinatória ocorre a união da aprendizagem subordinada e superordenada, o que torna o conhecimento mais sólido.

Fonte: (AUSUBEL, 2000).

Inspirados nas três categorias de aprendizagem significativa de Ausubel (2000), as sete dimensões da Metodologia 7Cs foram divididas em três fases: Descoberta; Aperfeiçoamento; e Consolidação. No Quadro 10 a seguir apresentamos as fases da metodologia, com suas descrições, as dimensões que cada uma contempla, o que esperamos durante a aplicação e a categoria de aprendizagem correspondente.

Quadro 10 - Fases da Metodologia 7Cs

METODOLOGIA 7CS				AUSUBEL (2000)
FASE	DESCRIÇÃO	DIMENSÕES	O QUE SE ESPERA	CATEGORIA DE APRENDIZAGEM
Descoberta	Consiste no contato inicial dos alunos quando iniciam a interagir com os conceitos e estruturas básicas de algoritmos.	<ul style="list-style-type: none"> ▪ Compreender ▪ Conceber 	Por meio da compreensão e interpretação da estrutura e construção dos seus primeiros algoritmos, pretende-se que os alunos adquiriram conhecimentos prévios, base para próxima fase.	Subordinada
Aperfeiçoamento	Consiste em sensibilizar seus alunos em perceber características importantes para construir um algoritmo com coerência e de fato executável.	<ul style="list-style-type: none"> ▪ Completar ▪ Compatibilizar ▪ Corrigir 	É o momento em que a partir dos subsunções adquiridos na fase anterior, os alunos irão concretizar, a solução dos problemas propostos nas dimensões da fase Aperfeiçoamento, a partir de seus conhecimentos preexistentes.	Superordenada
Consolidação	Esta fase, consiste em consolidar e combinar os conhecimentos adquiridos, para a construção de algoritmos eficientes.	<ul style="list-style-type: none"> ▪ Construir ▪ Criar 	É nesta fase que os alunos irão combinar os conhecimentos aprendidos nas duas primeiras fases, para solucionar problemas com maior grau de exigência e/ou complexidade, tornando o conhecimento mais sólido.	Combinatória

Fonte: Elaborado pelo autor.

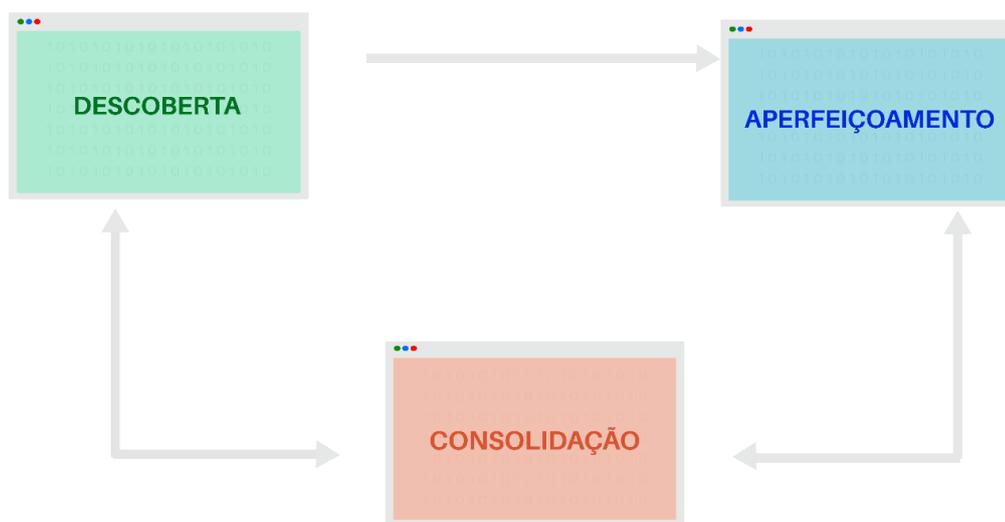
A primeira fase Descoberta é constituída por duas dimensões: Compreender e Conceber. Essa fase inicia com a primeira dimensão Compreender, na qual os alunos irão realizar a análise de trechos e/ou algoritmos simples completos, em busca de estimular a capacidade de interpretar e compreender as funcionalidades das estruturas de algoritmos trabalhadas, pelo professor. A primeira dimensão é importante para próxima, Conceber, pois é a partir da compreensão da estrutura de algoritmos ministrada pelo professor, que os alunos devem conceber seus primeiros algoritmos.

Na segunda fase, Aperfeiçoamento, os alunos devem aplicar os conhecimentos adquiridos na primeira fase. Esta fase (Aperfeiçoamento) compreende: a dimensão Completar, que tem objetivo de verificar a partir de um algoritmo já construído, elementos que precisam ser inseridos; a dimensão Compatibilizar, que consiste em organizar um algoritmo concebido de forma desorganizada, para que se torne executável; e a dimensão Corrigir, consiste na correção de algoritmos já construídos, que apresentam erros.

A terceira e última fase é Consolidação, na qual os alunos precisam unir todas as habilidades e competências e conhecimentos adquiridos nas fases anteriores, em um único objetivo, a solução de problemas, para construir algoritmos com eficiência e validade. Esta fase compreende: a dimensão Construir, na qual os alunos realizarão algoritmos com um nível de exigência maior; e a dimensão Criar, que se configura na construção de comandos com questões com situações problemas criadas pelos alunos, de forma colaborativa. As atividades colaborativas segundo Moreira (2011, p. 50) têm um grande potencial para facilitar a aprendizagem significativa em que viabiliza a troca de informações, a negociação dos significados e colocam a professor como mediador.

Ausubel (2000) afirma que a aprendizagem significativa ocorre mediante a ancoragem entre as ideias e conhecimentos anteriores relevantes, presentes na estrutura cognitiva dos indivíduos, como forma de adquirir e reter novos conhecimentos. Nesse sentido, pensamos na divisão em três fases, na qual ocorre a solidificação de conhecimentos específicos que consideramos importantes para construção e elaboração de algoritmos, para que pudéssemos elencar elementos por meio da ancoragem de conhecimentos, adquiridos de uma fase para outra, como ilustrado na Figura 2 a seguir.

Figura 2 – Fases da Metodologia 7Cs
ANCORAGEM DE CONHECIMENTOS



Fonte: Elaborado pelo autor

Na teoria de Ausubel (2000) os indivíduos aprendem melhor quando o novo conhecimento é relacionado com outros preexistentes. Por isso, ao partirmos do princípio de que a maioria dos alunos, dificilmente possui conhecimentos prévios sobre os conteúdos relacionados à construção de algoritmos, criamos um caminho metodológico para permitir que, a cada fase, o aluno possa adquirir conhecimentos prévios, que sirvam como ancoras para a fase seguinte.

A Figura 2 demonstra que a primeira fase (Descoberta) serve de âncora para a segunda (Aperfeiçoamento) e as duas (tanto a primeira quanto a segunda) para a terceira (Consolidação). A ancoragem de conhecimentos entre as fases é um ponto importante para o desenvolvimento da aprendizagem significativa dos alunos. Além disso, a forma de disposição das dimensões da Metodologia 7Cs, corrobora para sua aplicação na disciplina de algoritmos ao considerar a quantidade de aulas para cada estrutura de algoritmos. Propomos a seguir alguns exemplos de aplicação da metodologia em forma de aulas.

4.2 EXEMPLO DE APLICAÇÃO DA METODOLOGIA 7Cs

Nesta subseção apresentaremos exemplo de aplicação da Metodologia 7Cs para visualizar como poderá ser desenvolvida em disciplinas de algoritmos. Para tal, exemplificamos em forma de aulas. Com base em pesquisa documental tomamos como

base o plano de ensino da disciplina da turma do primeiro semestre de 2018 do curso graduação em Bacharelado em Ciência da Computação, da UFPA, para definição das aulas.

Aula 1 - Pontapé Inicial: Apresentação lúdica da Metodologia 7Cs e de conceitos básicos de algoritmos.

Julgamos ser importante a familiarização dos alunos com a Metodologia 7Cs, antes da sua aplicação, visto que isso conduzirá a forma prática da disciplina. Ao buscar tornar a proximidade com a metodologia mais prazerosa decidimos por desenvolver um jogo com o propósito de apresentar ludicamente a Metodologia 7Cs aos alunos da disciplina de algoritmos. Nomeamos o jogo de M7C, com base na concepção da metodologia, em que contempla cada uma das dimensões por meio de desafios com situações problemas. O jogo foi pensado para que pudesse de uma maneira divertida, ser apresentada a metodologia, e com ele trabalhar alguns conceitos básicos de algoritmos, no primeiro dia de aula.

O jogo será estruturado em forma de tabuleiro (Figura 3) e contém sete territórios. Cada território representa uma dimensão da Metodologia 7Cs, são elas Compreender, Conceber, Completar, Compatibilizar, Corrigir, Construir e Criar. No assim como na metodologia a ação Colaborar está presente em todo o jogo, não constituindo, portanto, um dos territórios. Dessa maneira, para realizar a atividade, os alunos devem ser divididos em grupos para resolver as situações problemas de forma colaborativa.

Figura 3 - Tabuleiro do jogo M7C

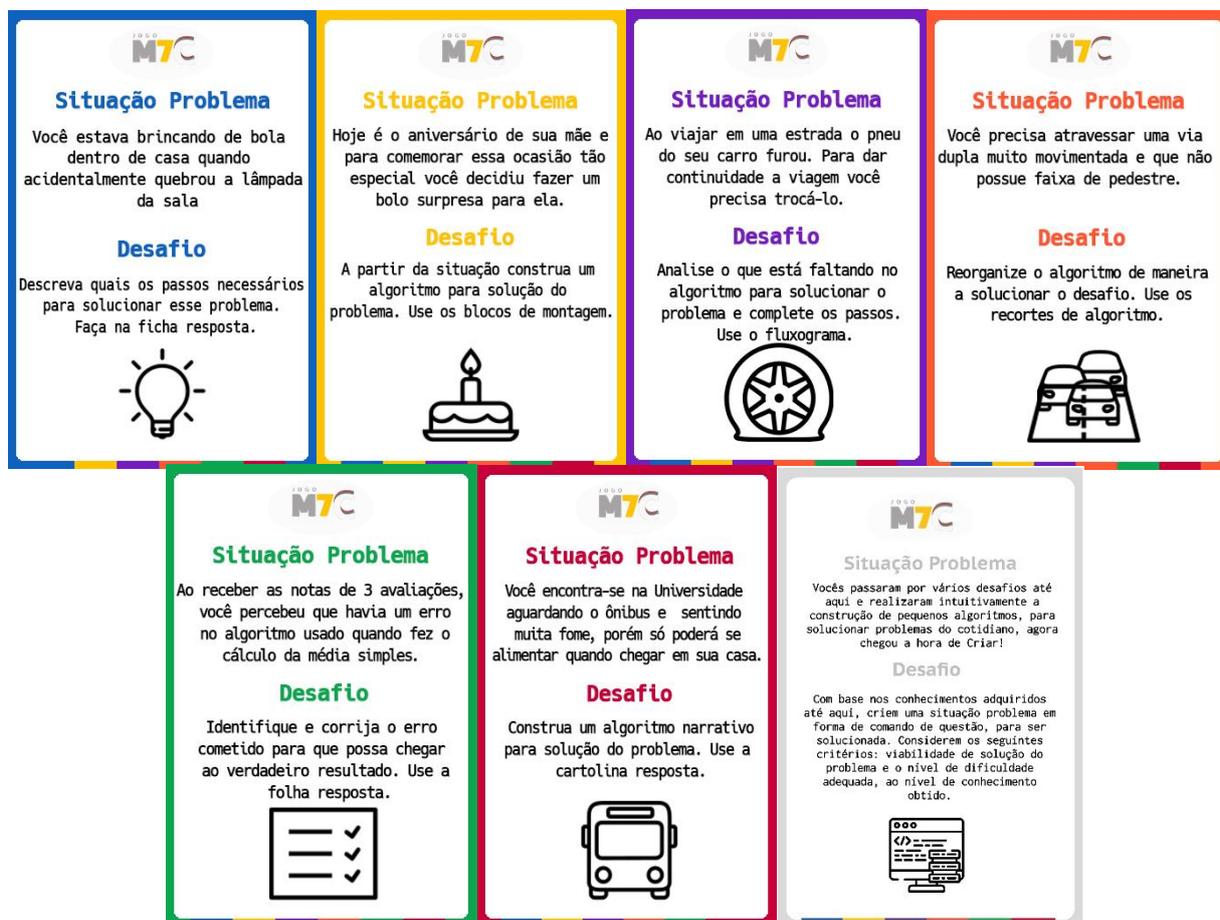


Fonte: Elaborado pelo autor.

São distribuídas diferentes situações problemas e desafios relacionados ao cotidiano para que os alunos compreendam determinados conceitos de algoritmos e para em cada território do jogo possa suscitar uma dimensão da Metodologia 7Cs. Ao iniciar o jogo todos os grupos formados irão realizar os desafios propostos ao mesmo tempo. O grupo que apresentar a melhor resposta marcará o território correspondente ao desafio realizado e vencerá aquele que tiver marcado mais territórios.

O objetivo é que os alunos, de maneira intuitiva, trabalhem a Metodologia 7Cs e alguns conceitos iniciais de algoritmos (como a estrutura sequencial, entrada e saída de dados, formas de representação, entre outras), proporcionando um aprendizado de maneira divertida, motivadora e significativa, ao possibilitar a interação e troca de conhecimentos e utilizar situações cotidianas que farão conexões entre conhecimentos preexistentes dos alunos e o novo conhecimento. Na Figura 4 são apresentadas as cartas do tipo Desafios.

Figura 4 - Cartas Desafio do jogo M7C



Fonte: Elaborado pelo autor

Durante o jogo o professor fica responsável por conduzir todo o processo a cada desafio realizado, deve analisar e verificar as respostas dadas pelos alunos, validando o grupo vencedor de cada uma das dimensões. Além disso, após cada desafio deve realizar o feedback para os alunos sobre a dimensão correspondente à Metodologia 7Cs e sobre os conteúdos iniciais trabalhados.

Na descrição das aulas seguintes faremos um exemplo de como pensamos em abordar a Metodologia 7Cs em uma das estruturas de algoritmos, a Estrutura de Seleção. As demais aulas apresentadas a seguir, faram um análogo para aplicação e adaptação da metodologia para as demais estruturas.

Aula 2 - Descoberta

Como já relatado, essa fase consiste no primeiro contato dos alunos com um novo conhecimento, ou seja, no início de cada estrutura de algoritmos. Como as estruturas tem particularidades, portanto se configura, em nosso entendimento, como um novo conhecimento. O professor pode escolher em qual estrutura ele quer aplicar a metodologia, ou, realizar a aplicação em todas se assim desejar. É importante ressaltar que antes de iniciar a fase, os alunos já devem saber os conceitos básicos sobre a estrutura de algoritmos que devem trabalhar.

Após conceituar e explicar as especificidades dos comandos da estrutura de seleção, os alunos devem passar pelas dimensões da Metodologia 7Cs. As dimensões presentes nesta fase são: Compreender e Conceber. A seguir na Aula 2 será mostrado um exemplo de aplicação das referidas dimensões, na estrutura de Seleção.

Compreender

Após aprenderem conceitualmente o conteúdo ministrado, é importante que os alunos compreendam as funcionalidades da estrutura a ser trabalhada. Para tal, elaboramos a dimensão Compreender em que os alunos deverão relatar de maneira escrita o que compreendem sobre um determinado trecho ou de um algoritmo inteiro apresentado. As habilidades e competências que pretendemos desenvolver nos alunos são:

- Compreender e interpretar os funcionamentos da estrutura de um algoritmo;
- Desenvolver a capacidade de solução de problemas.

É relevante levar em consideração, as seguintes indicações:

- É importante que os algoritmos ou trechos destes sejam escolhidos cuidadosamente, pois é o primeiro contato dos alunos com a estrutura trabalhada;
- É interessante que os alunos realizem as atividades no papel, para que a aprendizagem seja mais construtiva à medida que os eles adquirem um maior nível de concentração.

Para ficar mais compreensível, deixamos aqui um exemplo de exercício no Quadro 11:

Quadro 11 - Exemplo de exercício da dimensão Compreender

Exercício: A partir dos trechos de algoritmos a seguir, descreva, de acordo com sua compreensão, o que será executado?	
Código	Resposta
<pre> Begin writeln('Digite um número:'); readln(num); if (num > 20) then writeln('Esse número é MAIOR que 20.') else writeln('Esse número é menor que 20.');</pre> End.	<hr/> <hr/> <hr/> <hr/>

Fonte: Adaptado da lista de exercício usada na disciplina Programação Estruturada de Computadores, durante observação participante no segundo semestre de 2018.

Para esta dimensão, os alunos receberão uma lista de exercícios com trechos de códigos para que possam relatar suas impressões e o que compreenderam dos mesmos. Partimos da premissa de que, ao realizarem a atividade, os alunos desenvolverão habilidades e competências importantes para a construção de um algoritmo, são elas: A análise, compressão e interpretação dos funcionamentos da estrutura de um algoritmo.

Conceber

Após praticar a compreensão sobre a estrutura de seleção, os alunos passarão para a dimensão Conceber. Esta dimensão se trata do momento em que serão desenvolvidos os primeiros algoritmos, a partir de comandos determinados pelo professor. Com base no

comando, os alunos devem construir no papel a solução do problema em forma de algoritmo e posteriormente executar na linguagem de programação que o professor utilizará em suas aulas. As habilidades e competências que pretendemos desenvolver nos alunos são:

- Interpretar os comandos das questões;
- Desenvolver a capacidade de solução de problemas.

É relevante levar em consideração, as seguintes indicações:

- É interessante que os alunos possam fazer as soluções primeiramente no papel, para depois passarem para o computador. Assim, eles trabalham menos com tentativas e erros, como o computador permite fazer;
- Uma boa técnica que pode ser interessante ser estimulada nessa dimensão é o teste de mesa, o qual permite simular o comportamento do código construído no papel, para buscar identificar erros mesmo sem o computador.

O teste de mesa é realizado de maneira manual para validar a coerência lógica de um algoritmo construído. Para ficar mais compreensível, deixamos aqui um exemplo de exercício no Quadro 12:

Quadro 12 - Exemplo de exercício da etapa Conceber

Exercício: A partir dos comandos de algoritmos a seguir, conceba a solução dos seus algoritmos.
1) Escreva um algoritmo para ler um valor inteiro (do teclado) e escrever (na tela) o seu antecessor.
2) Faça um algoritmo que leia três notas de um aluno, calcule e escreva a média final desse aluno. Considerar que a média é ponderada e que o peso das notas é 2, 3 e 5.

Fonte: Adaptado da lista de exercício usada na disciplina Programação Estruturada de Computadores, durante observação participante no segundo semestre de 2018.

Consideramos ser importante que primeiro os alunos pratiquem no papel para posteriormente executar no computador. Essa prática contribui para que alunos de apropriem melhor dos comandos e suas funcionalidades de maneira mais concreta, ao força-los a simularem o comportamento do código construído e realizarem o teste de mesa, para posteriormente verificar na linguagem de programação.

Para atividade será disponibilizada uma lista de exercício para os alunos. Ao passarem pela dimensão Compreender os alunos podem ser capazes de realizar de

maneira mais ágil e coerente seus primeiros algoritmos. Na atividade buscamos desenvolver habilidades e competências como: interpretar os comandos das questões e desenvolver a capacidade de solução de problemas.

Aula 3: Aperfeiçoamento

A fase Aperfeiçoamento consiste na aplicação dos conhecimentos adquiridos nas dimensões anteriores (Compreender e Conceber), assim como aprender a importância de determinados aspectos para construção de um algoritmo como: inserir todos os elementos necessários para solução de um problema, ordenar de maneira coerente os comandos e identificar erros que podem influenciar no resultado a ser alcançado. Esta fase compreende as dimensões Completar, Compatibilizar e Corrigir, exemplificadas nas aulas a seguir.

Na aula anterior trabalhou-se a fase Descoberta em que os alunos passaram por um processo de construção do conhecimento acerca da estrutura de seleção por meio da interpretação e análise de trechos de algoritmos e concepção de seus primeiros códigos. Somente com os conhecimentos desenvolvidos anteriormente os alunos possuíam uma melhor para solucionar as atividades da fase Aperfeiçoamento. Para esta fase serão trabalhadas as dimensões Completar, Compatibilizar e Corrigir, descritas abaixo.

Completar

Nesta dimensão os alunos devem ser capazes de identificar com base no comando da questão e nos algoritmos disponibilizados, quais elementos estão faltando para que o código apresentado possa alcançar a solução desejada para o problema exposto (no comando da questão). As habilidades e competências que pretendemos desenvolver nos alunos são:

- Interpretar os comandos das questões e códigos;
- Desenvolver a capacidade de percepção de erros e de ajuste de algoritmo.

É relevante levar em consideração, as seguintes indicações:

- É importante que sejam escolhidos trechos de algoritmos bem elaborados, de forma a aumentar a probabilidade de sucesso dos alunos na atividade;

- Mais uma vez, acreditamos que os alunos devam trabalhar primeiramente no papel, realizando o teste de mesa.

Para ficar mais compreensível, deixamos aqui um exemplo de exercício no Quadro 13. A coluna resposta esperada apresenta o que esperamos que os alunos respondam. Na lista ela estará sem a resposta.

Quadro 13 - Exemplo de exercício da dimensão Completar

Exercício: A partir do comando, analise e identifique o que falta para que o algoritmo possa cumprir com o que está sendo solicitado.	
Comando da questão: Construir um algoritmo que leia dois números e efetue a adição. Caso o valor somado seja maior que 20, este deverá ser apresentado somando-se a ele mais 8; caso o valor somado seja menor ou igual a 20, este deverá ser apresentado subtraindo-se 5.	
Código Incompleto	Resposta Esperada
<pre> Program_Completar; //Declaração de variáveis var a, b, soma: integer; Begin writeln('Digite um valor para B:'); readln(b); if (soma > 20) then writeln('O valor da A + B é maior que 20, resultado final: ',soma +8) writeln('O valor de A + B é menor ou igual a 20, resultado final: ',soma - 5) else writeln('Valores invalidos'); readln; end. </pre>	<pre> Program_Completar; //Declaração de variáveis var a, b, soma: integer; Begin writeln('Digite um valor para A:'); readln(a); writeln('Digite um valor para B:'); readln(b); soma:= a + b; if (soma > 20) then writeln('O valor da A + B é maior que 20, resultado final: ',soma +8) else if (soma <= 20) then writeln('O valor de A + B é menor ou igual a 20, resultado final: ',soma - 5) else writeln('Valores invalidos'); readln; end. </pre>

Fonte: Adaptado da lista de exercício usada na disciplina Programação Estruturada de Computadores, durante observação participante no segundo semestre de 2018.

Para esta dimensão é necessário que os alunos inicialmente interpretem o comando da questão, para posteriormente analisarem o código e identificar quais elementos essenciais que estão faltando inserir para tornar a solução do problema coerente e executável. Este exercício busca desenvolver habilidades e competências como: interpretar os comandos das questões e códigos, desenvolver a capacidade de percepção de erros e de ajuste de algoritmo.

Compatibilizar

Percebemos durante a observação participante que os alunos possuíam algumas dificuldades de organização e encadeamento dos comandos em uma ordem lógica e coerente com a linguagem de programação ao qual estavam construindo seus algoritmos. Assim, pensamos na construção da dimensão Compatibilizar, em que os alunos devem analisar e interpretar o problema a ser solucionado, para em seguida reordenar as linhas de comando, de modo a encadear os passos de forma lógica, corretamente. As habilidades e competências que pretendemos desenvolver nos alunos são:

- Interpretar os comandos das questões;
- Desenvolver a capacidade de organizar o encadeamento das ideias;
- Desenvolver a capacidade de ajuste de algoritmo.

É relevante levar em consideração, as seguintes indicações:

- A dimensão Compatibilizar pode ser feita em forma de dinâmica em que os alunos recebem um algoritmo recortado para organizar de acordo com a solução do problema disponibilizado pelo professor;
- Para os alunos terem uma melhor visualização da proposta, o professor pode projetar o comando e o algoritmo desordenado.
- Por ser um código já inteiro, os alunos podem fazer diretamente no computador, para que possam testar e verificar os erros de ordenamento nas linhas de comando.

Para ficar mais compreensível, deixamos aqui um exemplo de exercício no Quadro 14:

Quadro 14 - Exemplo de exercício da dimensão Compatibilizar

<p>Exercício: A partir do algoritmo a seguir, analise e interprete o comando da questão, realocando as linhas de comando de forma ordenada e coerente para que o algoritmo possa corresponder ao que se pede.</p>	
<p>Comando da Questão: Faça um programa que receba três notas, calcule a média e mostre se o aluno foi aprovado se sua média for maior que igual ou maior que 7, quanto precisa tirar em novo exame e reprovado se sua média for entre 0 e 3.</p>	
Código Incompleto	Resposta Esperada
<pre> Program_compatibilizar; Begin Var Nota1, Nota2, Nota3, Media, Nota_exame: Real; Begin Write (' digite a terceira nota: '); Readln(Nota3); Write(' Digite a primeira Nota: '); Readln(Nota1); Write(' Digite a segunda nota: '); Readln(Nota2); Writeln('media Aritmética: ',Media:); If (Media >=0) And (Media <3) Then Writeln('reprovado'); If (Media >=3) And (Media < 7) Then Media := (Nota1 + Nota2 + Nota3) / 3; Begin Writeln('exame '); Nota_exame := 12 - Media; Writeln ('deve Tirar Nota ',Nota_exame:4:2,' Para Para Ser Aprovado'); End; If (Media >= 7) And (Media < 10) Then Writeln('ap2rovado'); Readln; End. </pre>	<pre> Program_compatibilizar; Begin Var Nota1, Nota2, Nota3, Media, Nota_exame: Real; Begin Write(' Digite a primeira Nota: '); Readln(Nota1); Write(' Digite a segunda nota: '); Readln(Nota2); Write (' digite a terceira nota: '); Readln(Nota3); Media := (Nota1 + Nota2 + Nota3) / 3; Writeln('media Aritmética: ',Media:); If (Media >=0) And (Media <3) Then Writeln('reprovado'); If (Media >=3) And (Media < 7) Then Begin Writeln('exame '); Nota_exame := 12 - Media; Writeln ('deve Tirar Nota ',Nota_exame:4:2,' Para Ser Aprovado'); End; If (Media >= 7) And (Media < 10) Then Writeln('aprovado'); Readln; End. </pre>

Fonte: Adaptado da lista de exercício usada na disciplina Programação Estruturada de Computadores, durante observação participante no segundo semestre de 2018.

Como o objetivo é que os alunos possam reorganizar o código tornando ele executável, inicialmente as linhas de comandos de algoritmos apresentados devem estar fora do lugar. Para esta dimensão os alunos precisam analisar o código e identificar quais os comandos que estão fora do lugar e reordenar. Esta atividade tem por objetivo desenvolver habilidades e competências como: interpretar os comandos das questões, desenvolver a capacidade de organizar o encadeamento das ideias e de ajuste de algoritmo.

Corrigir

Na estrutura de um algoritmo é fundamental que todos os elementos em sua construção estejam corretos para que possa ser executado e alcançar a solução do problema proposto. Esta dimensão consiste em identificar os erros em algoritmos já construídos, para então corrigi-los. Diferente da dimensão Compatibilizar, os erros serão em termos sintaxe e troca de elementos, como por exemplo, erros na escrita de comandos importantes e troca de operadores. As habilidades e competências que pretendemos desenvolver nos alunos são:

- Corrigir códigos com erros;
- Desenvolver a capacidade de análise e ajuste de algoritmos.

É relevante levar em consideração, as seguintes indicações:

- É interessante que primeiramente os alunos façam a correção dos algoritmos de maneira escrita;
- Após corrigir as questões no papel, podem também realizar o teste de mesa, para posteriormente executar os algoritmos na linguagem de programação utilizada pelo professor;
- O erro no código pode ser de sintaxe, com elementos trocados, como um sinal de mais (“+”) no lugar do sinal de menos (“-”), entre outros.

Para ficar mais compreensível, deixamos aqui um exemplo de exercício no Quadro 15:

Quadro 15- Exemplo de exercício da dimensão Corrigir

Exercício: A partir do código a seguir, analise e interprete o comando da questão para identificar no código disponibilizado os possíveis erros e corrigi-los de forma a alcançar o que se pretende.	
Comando da questão: Desenvolver um algoritmo que leia dois números e mostrar o valor do maior número ou dizer se os valores são iguais.	
Código Incompleto	Resposta Esperada
<pre> Program_Corrigir; Var Num1, Num2: Real; Begin Write ('Digite o primeiro número: '); Readln(Num1); Write('Digite o segundo número: '); Readln(Num2); If Num1 > Num2 then Writeln ('O maior número : ',Num2) Else If Num2 < Num1 then Writeln('O maior número : ', Num3) Else Writeln('Os números são iguais'); Readln; End. </pre>	<pre> Program_Corrigir; Var Num1, Num2: Real; Begin Write ('Digite o primeiro número: '); Readln(Num1); Write('Digite o segundo número: '); Readln(Num2); If Num1 > Num2 then Writeln ('O maior número : ',Num1) Else If Num2 > Num1 then Writeln('O maior número : ', Num2) Else Writeln('Os números são iguais'); Readln; End. </pre>

Fonte: Adaptado da lista de exercício usada na disciplina Programação Estruturada de Computadores, durante observação participante no segundo semestre de 2018

Nesta dimensão é necessário que os alunos inicialmente analisem o código e identifiquem quais elementos estão inseridos de maneira errada, para corrigir os erros e tornar a solução do problema coerente e executável. Primeiramente os alunos realizarão de maneira escrita a correção de algoritmos já construídos e depois poderão executá-los na linguagem de programação utilizada pelo professor. As habilidades e competências que pretendemos desenvolver são: corrigir códigos com erros, desenvolver a capacidade de análise de algoritmos e de ajuste de algoritmo.

Aula 4 - Consolidação

Nesta fase os discentes deverão a partir de um comando, desenvolver a solução de problemas com a aplicação de todos os conhecimentos adquiridos anteriormente. Esperamos que após as fases anteriores os alunos desenvolvam habilidades e competências essenciais para a construção de algoritmos com um maior grau de complexidade. Nas aulas anteriores os alunos passaram por um processo de construção de conhecimentos e o desenvolvimento de habilidades e competências essenciais para

construção de algoritmos com eficiência. A fase Consolidação compreende as dimensões Construir e Criar, exemplificadas a seguir.

Construir

Para realização da dimensão Construir, os discentes receberão uma lista de exercício que conterá comandos para solução de determinados problemas. É importante ressaltar que a diferença entre as dimensões Conceber e Construir está no nível de complexidade dos exercícios. Na dimensão Conceber pensamos no “nascimento” dos primeiros algoritmos realizados pelos alunos de maneira simples, já na dimensão Construir se consolida todos os conhecimentos adquiridos anteriormente na solução de problemas que exigem maior esforço e conhecimento dos alunos. As habilidades e competências que pretendemos desenvolver nos alunos são:

- Interpretar os comandos das questões;
- Desenvolver a capacidade de solução de problemas;
- Construir algoritmos com eficiência.

É relevante levar em consideração, as seguintes indicações:

- Os alunos praticarem primeiramente no papel e posteriormente na linguagem de programação, para melhor apropriação da construção e encadeamento das informações de um algoritmo;
- Os professores forneçam um *feedback* das atividades realizadas pelos alunos.

Para ficar mais compreensível, deixamos aqui um exemplo de exercício, no Quadro 16:

Quadro 16 - Exemplo de exercício da dimensão Construir

Exercício: A partir do comando disponibilizado e dos aprendizados alcançados até o presente momento, construa a solução do problema.	
Comando da questão: Escrever um algoritmo que lê um conjunto de 4 valores i, a, b, c, onde i é um valor inteiro e positivo e a, b, c, são quaisquer valores reais e os escreva. A seguir: a) Se i=1 escrever os três valores a, b, c em ordem crescente. b) Se i=2 escrever os três valores a, b, c em ordem decrescente. c) Se i=3 escrever os três valores a, b, c de forma que o maior entre a, b, c fique dentre os dois.	
Resposta esperada	
<pre> Program_ Construir; var n1,n2,n3,primeiro,segundo,terceiro,aux,i: integer; Begin writeln('Informe os tres numeros:'); Readln(n1,n2,n3); primeiro := n1; segundo := n2; terceiro := n3; if (primeiro > segundo) then begin aux:= primeiro; primeiro:= segundo; segundo:= aux; end; if (primeiro > terceiro) then Begin aux:= primeiro; </pre>	<pre> primeiro:= terceiro; terceiro:= aux; end; if (segundo > terceiro) then Begin aux:= segundo; segundo:= terceiro; terceiro:= aux; end; Writeln ('Digite a opção desejada:'); Readln(i); Case i of 1 : writeln('Resultado: ',primeiro,' ', segundo,' ', terceiro); 2 : writeln('Resultado: ',terceiro,' ', segundo,' ', primeiro); 3 : writeln('Resultado: ',segundo,' ', terceiro,' ', primeiro); End; end. </pre>

Fonte: Adaptado da lista de exercício usada na disciplina Programação Estruturada de Computadores, durante observação participante no segundo semestre de 2018

O objetivo da dimensão é que os alunos possam praticar exercícios com um maior grau de dificuldades e desenvolver habilidades e competências como: interpretar os comandos das questões, desenvolver a capacidade de solução de problemas e construir algoritmos com eficiência. Assim como na dimensão Conceber, indicamos que os alunos pratiquem primeiro no papel e posteriormente na linguagem de programação, para melhor apropriação da construção e encadeamento das informações de um algoritmo.

Como referido anteriormente nessa fase da metodologia os alunos devem aplicar as habilidades e competências desenvolvidas durante o percurso. Na próxima dimensão iremos buscamos potencializar os conhecimentos adquiridos por meio do estímulo do processo criativo a ação de colaborar, dos alunos.

Criar

A dimensão Criar consiste em estimular a solução de problemas e na promoção do estímulo ao processo criativo e colaboração entre os alunos, assim como potencializar a aprendizagem significativa por meio da troca de conhecimentos entre os alunos e a resolução de problemas. Esta dimensão, assim como a anterior (Construir), requer que os alunos utilizem os conhecimentos adquiridos até o momento. As habilidades e competências que pretendemos desenvolver nos alunos são:

- Capacidade de trabalhar em grupo;
- Aprendizado de maneira colaborativa;
- Potencialização da solução de problemas;
- Estimulação do processo criativo.

É relevante levar em consideração, as seguintes indicações:

- Para a formação dos grupos, se possível, é interessante mesclar alunos que têm mais facilidade na construção de algoritmos com os que têm menos;
- Vocês, professores(as), podem auxiliar os alunos em dúvidas que não estejam relacionadas à solução do problema de forma direta;
- Os comandos das questões podem ser feitos em folhas destacadas para facilitar na hora de passar para a outra equipe após o sorteio;
- Ao validar as respostas, é interessante chamar um representante do grupo que criou o comando da questão para verificar se a solução solicitada foi alcançada;
- Se dispuser de tempo suficiente, peçam que os alunos apresentem a solução desenvolvida por eles como forma de disseminar o conhecimento;
- É importante observar se todos os alunos estão realmente interagindo e contribuindo para criar os comandos das questões, bem como a solucionar os problemas após o sorteio.

Para ficar mais compreensível, deixamos aqui um exemplo de exercício. Para esta dimensão propomos uma dinâmica, com dois momentos, apresentados a seguir.

1º momento

- Explicar e apresentar a atividade, seu objetivo e como será desenvolvida.
- Dividir a sala em grupos.
- Lançar o desafio para que os alunos criem um comando de questão com uma situação problema, levando em consideração os seguintes requisitos:
 - a. O comando da questão deve ser elaborado com a participação de todos os integrantes do grupo.
 - b. Criar um problema viável, para ser solucionado.
 - c. Deve ter um nível de complexidade, de acordo com os conhecimentos adquiridos até o momento.

Acreditamos que o tempo de 30 min é suficiente para os grupos laborarem os comandos das questões.

2º Momento

- Após a elaboração dos comandos das questões, um novo desafio. Os grupos devem ser numerados pelo professor, e deve realizar um sorteio respeitando a seguinte condição:
 - a. Se o número retirado for da própria equipe, devolva para o sorteio.
 - b. Se o número for diferente da sua equipe, então deverá solucionar o problema proposto.

Os grupos devem iniciar o desafio ao comando do professor e ao terminarem a solução do problema sorteado, devem executar o código construído, para validar a resposta. O professor será o mediador durante a execução do desafio para auxiliar os alunos em dúvidas que não estejam relacionadas a solução do problema de forma direta, assim como também será o responsável por avaliar e confirmar se a resposta certa está coerente com o que foi pedido ou não.

O objetivo é que a atividade proporcione um ambiente interativo e de contribuição entre os alunos para integração e compartilhamento dos conhecimentos ao estimular discussões e reflexões sobre a construção de um algoritmo. Buscamos desenvolver nesta

dimensão habilidades e competências como: a capacidade de trabalhar em grupo, aprender de forma colaborativa, potencializar a solução de problemas, aperfeiçoar algoritmos e estimular o processo criativo.

5 APLICAÇÃO DA METODOLOGIA 7Cs

Neste capítulo explicitaremos como se deu a aplicação, na Universidade Federal do Pará (UFPA), da Metodologia 7Cs em dois diferentes momentos do ano de 2019: (i) Curso de Nivelamento de Algoritmos, considerado como Teste Piloto; e (ii) na disciplina Algoritmos no Curso de Bacharelado em Ciência da Computação (CBCC). Em ambas as situações, manifestaremos nossas considerações sobre as impressões que tivemos durante sua execução em sala de aula.

5.1 TESTE PILOTO DA METODOLOGIA 7Cs

Para o teste piloto da Metodologia 7Cs, realizamos uma parceria em março de 2019 com o Programa de Nivelamento em Algoritmos (PNA) no Instituto de Ciências Exatas e Naturais (ICEN) da UFPA. O PNA é em uma iniciativa discente, cujo objetivo é a introdução de conteúdos básicos de algoritmos, por meio de um curso de nivelamento, como forma de auxiliar a aprendizagem dos calouros do CBCC/ICEN quando do início da graduação. O desenvolvimento do PNA ocorreu mediante a participação de monitores, discentes do CBCC/ICEN de diferentes semestres, responsáveis por ministrar as aulas.

Em nossa parceria, a proposta era o uso da Metodologia 7Cs na parte prática do curso, após a ocorrência das aulas teóricas de algoritmos. Assim, para melhor clareza aos monitores, realizamos uma reunião de apresentação da metodologia em que expusemos seus objetivos, as habilidades e competências que pretendíamos que os alunos desenvolvessem em cada fase e um exemplo de aplicação para inspirá-los quanto aos exercícios. Realizamos, assim, um momento de instrução, de maneira que os monitores compreendessem a essência e a importância de cada dimensão proposta na metodologia. Os monitores demonstraram interesse e empolgação com o que foi apresentado e se prepararam para seu uso.

O curso ocorreu no período de 12/03 a 15/03 de 2019, contemplando quatro dias com quatro aulas diárias. Contou com a participação de 23 calouros. A descrição desse teste é apresentada a seguir.

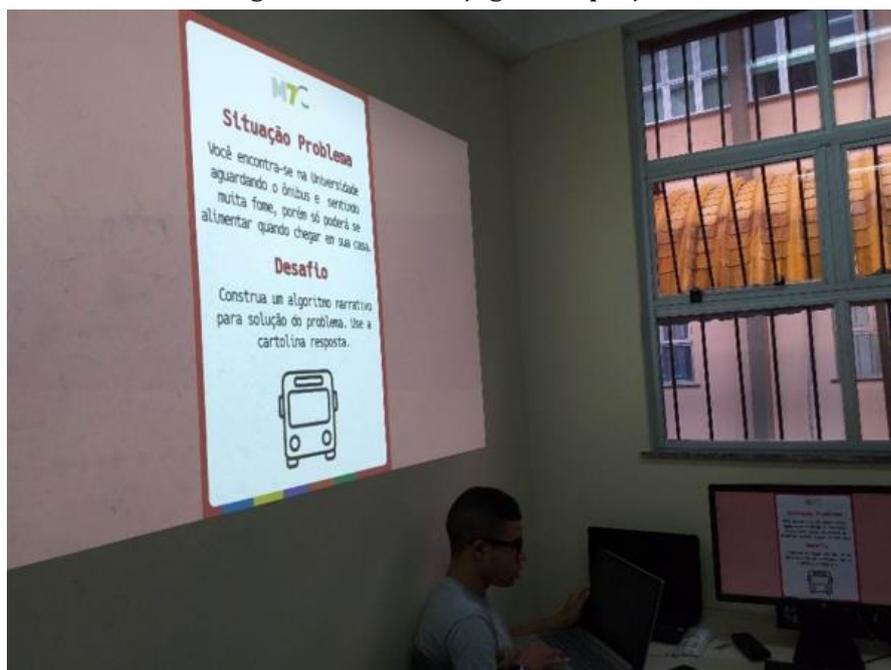
5.1.1 Apresentação da Metodologia 7Cs: teste do Jogo M7C

Como parte da aplicação, os alunos foram apresentados à metodologia, a fim de que se familiarizassem com suas particularidades e entendessem o porquê de seu uso. Assim, no primeiro dia os monitores iniciaram as aulas com a aplicação do jogo intitulado M7C, desenvolvido com a finalidade de apresentar a Metodologia 7Cs de maneira mais dinâmica e lúdica.

Para a aplicação do jogo, instruímos os monitores que primeiramente explicassem as regras aos discentes e a forma de condução. Em seguida, deveriam dividir a turma em grupos. Além de apresentar as dimensões da Metodologia 7Cs, o jogo M7C também trabalha conceitos iniciais de algoritmos (o que é um algoritmo; estrutura sequencial; formas de representação; estrutura de condição; e a construção de algoritmos narrativos).

Em sua proposta original, durante o jogo, as cartas devem ser disponibilizadas de maneira impressa e os discentes devem ter em mãos todos os materiais necessários para conseguir resolver as questões. Entretanto, por falta de recursos, os monitores precisaram inovar na aplicação do jogo. As cartas, por exemplo, foram apresentadas no projetor para que todos os alunos pudessem ver (Figura 5), e o tabuleiro foi desenhado no quadro magnético.

Figura 5 - Carta do jogo M7C projetada



Fonte: arquivo pessoal.

Além disso, para cada desafio presente na carta, existe uma Folha-Resposta correspondente com elementos importantes para a solução dos problemas, que também foi projetada. Nesse caso, a partir do modelo projetado, os discentes resolveram as questões em folhas de papel em branco, como mostra a Figura 6. Interessante ressaltar que a ausência do material impresso não prejudicou o andamento da atividade; ao contrário, demonstrou que o jogo é flexível e, ainda assim, alcança o objetivo pretendido, qual seja, apresentar a Metodologia 7Cs e trabalhar conceitos iniciais de algoritmos.

Figura 6 - Resolução dos desafios de maneira escrita



Fonte: arquivo pessoal.

Com relação aos desafios propostos pelas cartas, foram várias as formas utilizadas durante as aulas para chegar às soluções. O desafio referente à dimensão Compatibilizar, por exemplo, consistia em colocar em ordem um código de algoritmo que estava desorganizado, isto é, sem um encadeamento lógico. O código de algoritmos foi projetado na tela e, para a solução do problema, disponibilizamos aos discentes recortes de EVA simbolizando as linhas de código. Os discentes tiveram que escrever uma linha de código em cada pedacinho de EVA e remontar o algoritmo desorganizado projetado, agora com o sequenciamento correto dos passos (Figura 7).

Figura 7 - Grupo discutindo sobre a solução do desafio (dimensão Compatibilizar)



Fonte: arquivo pessoal.

Durante a aplicação do jogo, os monitores foram os responsáveis por conduzir todo o processo. A cada desafio, analisavam a solução dada pelos discentes de forma a verificar se esta estava coerente com o desafio proposto. Ou seja, os monitores validaram em todos os desafios de cada uma das dimensões as respostas dos grupos vencedores. A Figura 8 retrata os monitores realizando a análise da resposta do desafio da dimensão Compatibilizar.

Figura 8 - Análise das respostas do desafio realizada pelos monitores (dimensão Compatibilizar)



Fonte: arquivo pessoal.

Todos os monitores participavam da avaliação das respostas. Lembramos que o jogo é constituído por territórios e em cada desafio deve existir um grupo vencedor. Para isso, os grupos, ao terminarem o desafio, acionavam os monitores, os quais verificavam as respostas corretas, e aquela que tivesse sido entregue primeiro determinava o grupo vencedor.

Após o desafio e a escolha do grupo vencedor, os monitores realizaram o *feedback* para os discentes sobre a dimensão correspondente à Metodologia 7Cs, indicando quais os conteúdos iniciais trabalhados. Na Figura 9, os monitores estão expondo duas soluções elaboradas pelos discentes, em que, como *feedback*, explicaram as razões pelas quais um grupo resolveu corretamente e o outro não. Destacaram, ainda, a importância de as linhas de código contendo os comandos estarem em uma ordem lógica para se alcançar uma solução coerente. Durante toda a aplicação, os monitores, de maneira interativa, procuravam demonstrar em cada desafio seu propósito e a que fase da Metodologia 7Cs pertencia, destacando a importância de cada dimensão para o aprendizado dos discentes.

Figura 9 - Monitores realizando *feedback* sobre as soluções dadas por dois grupos (desafio da dimensão Compatibilizar)



Fonte: arquivo pessoal.

A formação dos grupos para o jogo M7C objetiva proporcionar um ambiente colaborativo, em que todos os discentes desempenham um papel importante no processo de aprendizagem. Dessa maneira, a ação Colaborar é transversal aos desafios propostos no jogo, pois em cada um deles é essencial a participação de todos. Assim, a condução da metodologia permitiu a interação entre os discentes e entre eles e monitores. Essa troca de experiências e conhecimentos potencializa a aprendizagem, o que torna o processo,

conforme Moreira (2011), mais significativo. A Figura 10 retrata os alunos interagindo sobre a solução do desafio referente à dimensão Compatibilizar.

Figura 10 - Interação entre os alunos para a solução do desafio (dimensão Compatibilizar)



Fonte: arquivo pessoal.

Utilizar a ludicidade para introduzir os conceitos de algoritmos trouxe leveza e motivação aos alunos, facilitando a compreensão dos conteúdos. Os alunos mostraram-se dispostos e alegres em participar das atividades propostas, criando um ambiente favorável para a etapa seguinte do desenvolvimento propriamente dito das dimensões de cada fase prevista na Metodologia 7Cs.

Apesar da ocorrência de mudanças durante a aplicação, verificamos que o jogo, conforme já citado, conseguiu alcançar seu objetivo. Os grupos realizaram todos os desafios sem precisar utilizar as Cartas de Ajuda criadas no jogo. A aplicação do jogo no PNA serviu para verificarmos, durante sua execução, o que poderia ser aprimorado, suas potencialidades como um instrumento didático, além de identificar outras formas possíveis de aplicação em sala de aula.

5.1.2 Desenvolvendo as fases

O desenvolvimento de cada fase da Metodologia 7Cs ocorreu a partir do segundo dia de curso. Para isso, antes de iniciar sua aplicação, os monitores ministraram os conteúdos de maneira teórica e realizaram um processo de ambientação da linguagem de

programação *Python*, com pequenos exemplos de Estrutura de Seleção. Dessa maneira, os discentes ficaram preparados para a fase da Descoberta, a qual contempla as dimensões Compreender e Conceber.

Os monitores ficaram livres para encontrar a melhor forma de condução da Metodologia 7Cs para o momento da prática em sala de aula. Ao percebermos se os monitores conseguiriam aplicar a metodologia sem nossa interferência, poderíamos verificar o quanto eles se apropriariam da proposta e quais estratégias seriam utilizadas em cada uma das fases. A aula iniciou pela Estrutura de Seleção, pois os monitores acreditaram que o jogo M7C no primeiro dia de curso conseguiu suprir os conhecimentos dos discentes sobre Estrutura Sequencial. Após a explicação das funcionalidades e elementos necessários para a construção de algoritmos com a estrutura de seleção, os monitores iniciaram a aplicação da metodologia.

Para a dimensão Compreender, foram exibidos no projetor trechos de códigos de algoritmos para que os discentes pudessem expressar o que estavam entendendo do código. Inicialmente, os discentes deveriam resolver a atividade no papel seguido de um relato aberto sobre o que haviam interpretado ao analisar os trechos de códigos, o que promoveu a interação e o compartilhamento de pensamentos diversificados, tornando a aula mais interessante. Na primeira dimensão já foi possível percebermos o quanto a Metodologia 7Cs é flexível, podendo ser usada de diferentes maneiras, sem interferir em seu principal objetivo: nesse caso específico, a compreensão das estruturas presentes nos algoritmos.

Na dimensão Conceber, os discentes iniciaram a construção dos primeiros algoritmos com problemas considerados simples, relacionados à soma, subtração, divisão e média aritmética. Nesse momento, os alunos puderam contar com o auxílio e a orientação dos monitores, caso sentissem dificuldades, como mostra a Figura 11. Ainda assim, verificamos, por meio das observações realizadas durante a execução da dimensão Conceber, que a maioria dos alunos não sentiu dificuldade na solução dos problemas propostos.

Figura 11 - Construção dos primeiros algoritmos dos alunos no curso



Fonte: arquivo pessoal.

O fato de os alunos precisarem analisar e interpretar trechos sobre a estrutura de algoritmos a ser trabalhada pode ter possibilitado aos discentes maior facilidade na construção dos seus primeiros algoritmos nesse contato inicial. A fase de Descoberta se constituiu como conexão inicial com a solução de problemas por algoritmos e a descoberta de novos elementos por meio da prática, acrescentando à estrutura cognitiva dos alunos conhecimentos que serviram como âncoras importantes para o desenvolvimento de atividades mais complexas.

No terceiro dia de curso, resolvemos aplicar a Metodologia 7Cs de uma maneira diferente. Era o momento de pensar fora da caixinha. Para aplicação da segunda fase da Metodologia 7Cs, Aperfeiçoamento, com as dimensões Completar, Compatibilizar e Corrigir, colocamos aos monitores o desafio de utilizar o *Kahoot!*, que consiste em uma plataforma de aprendizagem baseada em jogos. Dessa maneira, os monitores criaram, para cada dimensão (Completar, Compatibilizar e Corrigir), dez questões no *Kahoot!* (Anexo I).

No *Kahoot!* existem três formas possíveis de desenvolver atividades: (i) modo Questionário, com perguntas e respostas; (ii) modo Desordem para montar a ordem correta da informação; e (iii) modo Pesquisa, em que podemos realizar levantamentos importantes sobre as impressões dos alunos em relação ao objeto da pesquisa, como a

Metodologia 7Cs, por exemplo. Na plataforma, quem cria as questões tem o acesso às perguntas e às respostas e torna-se o responsável por iniciar o jogo.

Após disponibilizarem aos discentes o número de identificação, foi possível acessar o *Kahoot!* nos computadores presentes no laboratório e iniciar a partida (Figura 12).

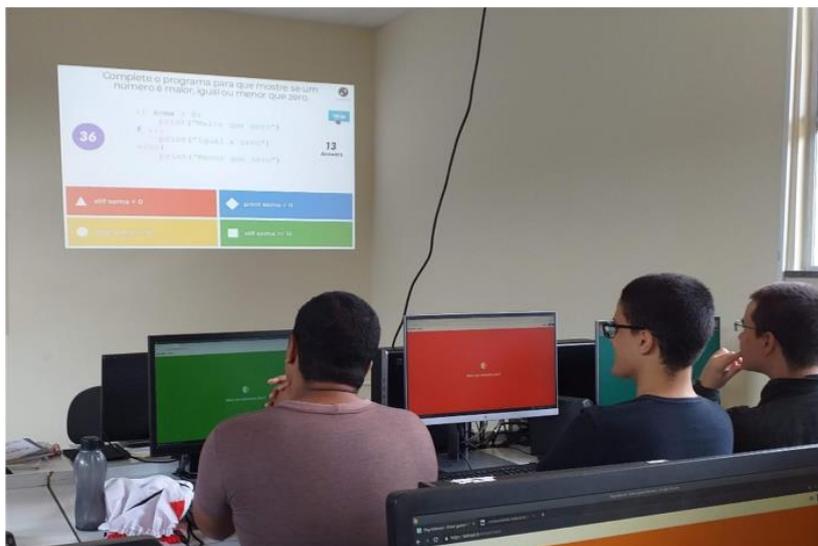
Figura 12 - Disponibilização do código de acesso do jogo



Fonte: arquivo pessoal.

Uma vez que somente no computador do usuário criador do jogo é que aparecem as perguntas com suas respectivas alternativas, os monitores projetaram o jogo na parede para que todos pudessem acompanhar (Figura 13).

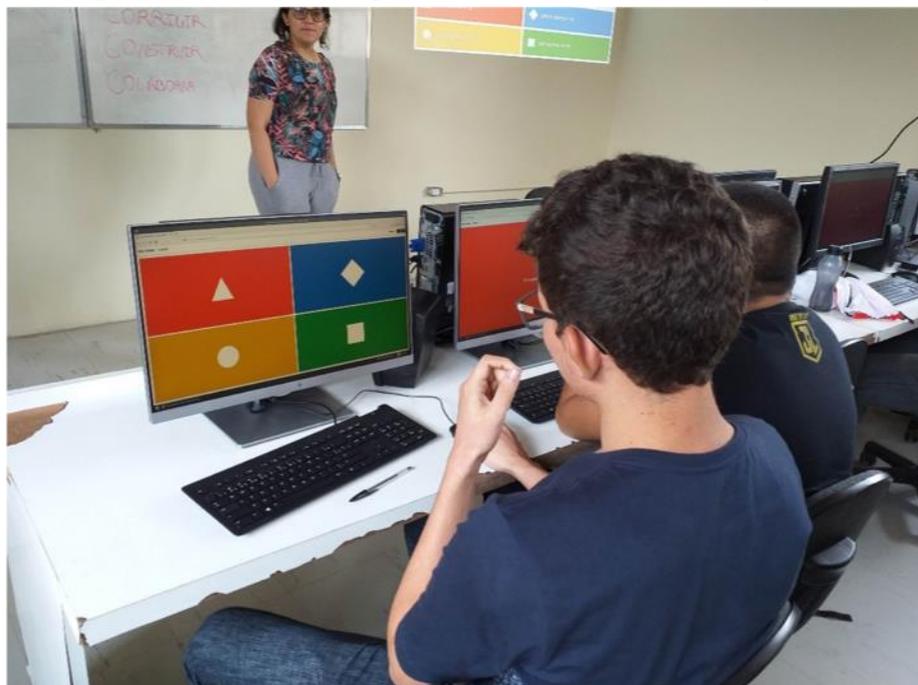
Figura 13 - Jogo do *Kahoot!* projetado para os alunos



Fonte: arquivo pessoal.

Cada alternativa de resposta possui uma cor específica. Assim, para cada pergunta projetada na parede, os discentes deveriam marcar a alternativa considerada correta em seus computadores, por meio da cor (Figura 14).

Figura 14 - Cores correspondentes aos alternativas da questões



Fonte: arquivo pessoal.

Para a fase Aperfeiçoamento, como mencionado anteriormente foram criadas dez perguntas, para cada dimensão. Para a dimensão Completar, as perguntas foram realizadas no modo Questionário, no qual foram inseridos códigos de algoritmo em que faltavam determinados elementos, para que os discentes identificassem, entre as alternativas de respostas, o elemento necessário para tornar coerente a construção do algoritmo. A Figura 15 apresenta um exemplo de questão da dimensão Completar.

Figura 15 - Cores correspondentes as alternativas da questões

Para completar o programa que demonstra números negativos e positivos, o que está faltando?

```
num = int
if num = 0:
    if num > 0:
        print("Positivo.")
    else:
        print("Negativo.")
else:
    print("Nulo.")
```

51

00
Respostas

Pular

▲ A linha 1 deve vir acompanhada de `(input ())`.

◆ Uma alternativa 1 e um ! na linha 2 para um número diferente de 0.

● Uma alternativa 1 e 2 e um : depois do 0 na linha 3.

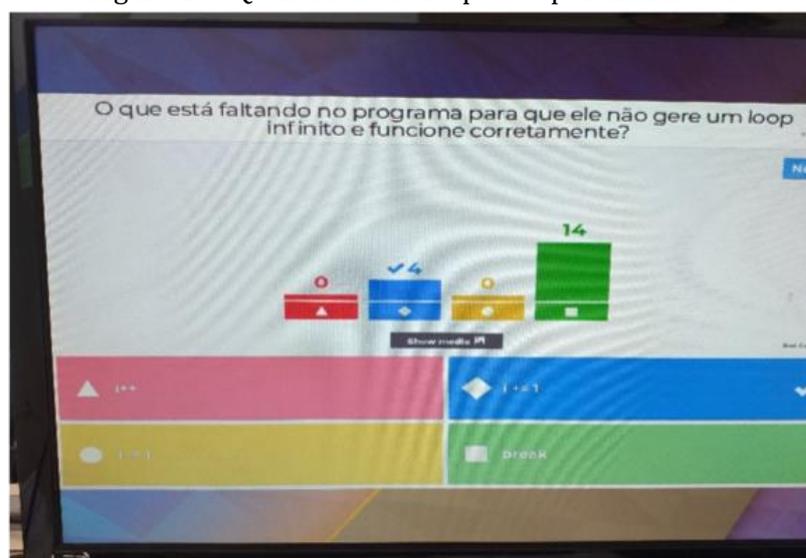
■ Não há nada para completar.

kahoot.it PIN do jogo: **596196**

Fonte: Atividade criada no *Kahoot!* pelos monitores, ministrantes do curso de nivelamento de algoritmos.

Em cada questão preparada no *Kahoot!*, os monitores determinaram um tempo de resposta dos jogadores. Na Figura 15, por exemplo, faltavam 51 segundos para a finalização das respostas. Quando todos os discentes marcavam suas respostas, aparecia projetada a quantidade de jogadores por alternativa escolhida, em que era possível identificar a quantidade de erros e certos (Figura 16).

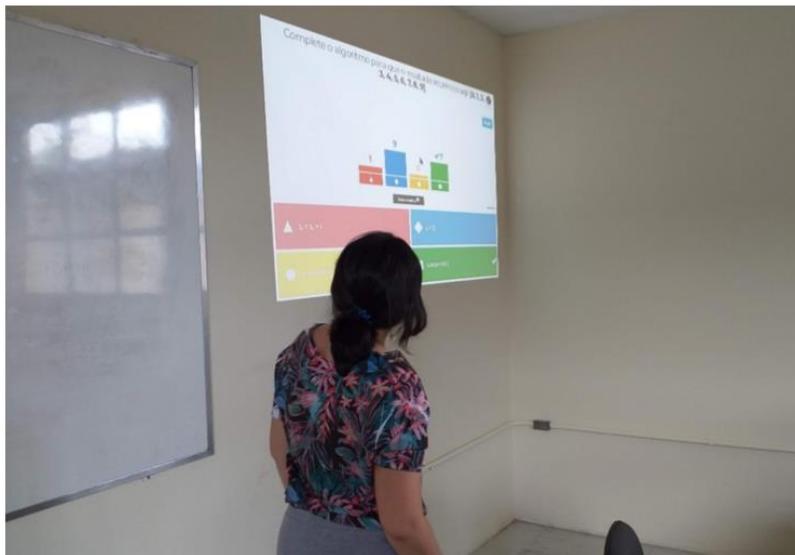
Figura 16 - Quantidade de respostas por alternativa



Fonte: arquivo pessoal.

Esse fato permitiu aos monitores discutir com os discentes algumas questões e lhes explicar o porquê da alternativa certa, buscando discutir as dúvidas presentes e esclarecê-las. De certa forma, tais resultados permitiram também verificar o nível de compreensão sobre os conteúdos ministrados durante o curso. A Figura 17 demonstra a monitora verificando o resultado das respostas dadas pelos discentes.

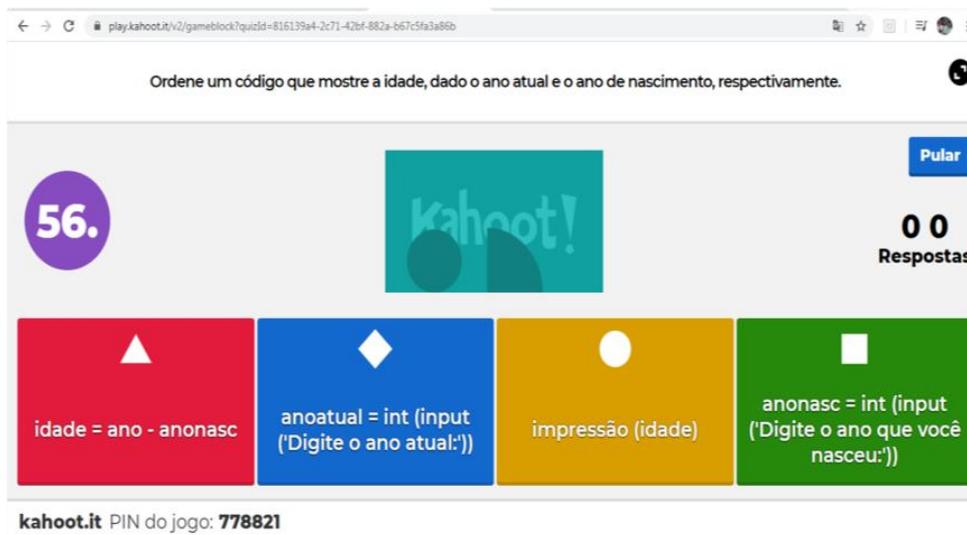
Figura 17 - Verificação da monitora sobre a quantidade de acertos e erros



Fonte: arquivo pessoal.

A dimensão Compatibilizar foi realizada no modo Desordem, com o objetivo de que os discentes pudessem colocar em ordem as linhas de trechos de códigos de algoritmo, para torná-los executáveis. Portanto, os monitores colocaram em cada alternativa uma linha de código de algoritmo em uma ordem correta, mas que ao iniciar o jogo foi embaralhada pelo próprio *Kahoot!*. A seguir, a Figura 18 apresenta um exemplo de questão da dimensão Compatibilizar, com linhas de códigos colocadas de maneira aleatória.

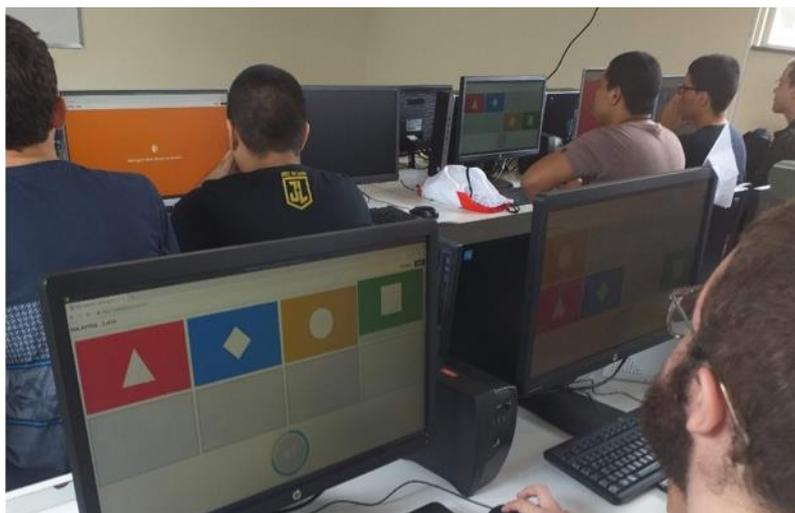
Figura 18 - Questão utilizada para a dimensão Compatibilizar



Fonte: Atividade criada no *Kahoot!* pelos monitores, ministrantes do curso de nivelamento de algoritmos.

O modo Desordem encaixou-se perfeitamente para contemplar o objetivo da dimensão Compatibilizar. Diferente do modo Questionário, os discentes, em vez de marcar as alternativas, teriam que montá-las na ordem correta, ou seja, em seus computadores, teriam que colocar as cores de maneira que as estruturassem na forma de um algoritmo e pensar na ordem lógica das linhas de códigos. A Figura 19 mostra os discentes tentando colocar as cores na ordem certa.

Figura 19 - Escolha da ordem das cores para alcançar a resposta correta



Fonte: arquivo pessoal.

Finalmente, para a dimensão Corrigir, as questões foram criadas no modo Questionário, para que os discentes pudessem verificar, em trechos de códigos de algoritmos, quais os erros presentes e escolher a alternativa que pudesse corrigir e solucionar os problemas. Para exemplificar melhor, na Figura 20 seguir apresentamos uma questão da dimensão Corrigir.



Fonte: Atividade criada no *Kahoot!* pelos monitores, ministrantes do curso de nivelamento de algoritmos.

O uso da ferramenta (*Kahoot!*) nos mostrou outra possibilidade de aplicação para a fase Aperfeiçoamento da Metodologia 7Cs, tornando a aula mais dinâmica e atrativa, com o cuidado de não perder de vista o fomento à aprendizagem significativa dos alunos.

Ainda no terceiro dia de curso realizamos a aplicação da terceira fase da metodologia, a Consolidação. Essa fase compreende as dimensões Construir e Colaborar. Lembramos que no teste piloto da Metodologia 7Cs, a dimensão Colaborar (retirada na versão final da proposta da Metodologia) ainda constava como uma das sete dimensões.

Para a dimensão Construir, os monitores passaram questões com um maior nível de complexidade, levando em consideração a proposta do curso de nivelar os conhecimentos de conteúdos básicos de algoritmos. As questões foram disponibilizadas em sala, e os discentes foram desafiados a construir os algoritmos de maneira individual e livres para resolver no papel, antes de testar no computador ou diretamente no computador.

Apesar de grande parte dos discentes terem tido o primeiro contato com os conteúdos básicos de algoritmos, percebemos que houve pouca dificuldade durante essa dimensão. Esse é um fato importante, pois, embora não tenhamos condições de afirmar que o bom desempenho dos discentes foi devido ao uso da Metodologia 7Cs, esta também não os prejudicou, ou seja, não interferiu negativamente no processo de aprendizagem dos conteúdos.

Para a dimensão Colaborar, a qual ainda constava como a última dimensão da Fase Consolidação, o objetivo era proporcionar uma atividade em que os discentes pudessem interagir e trabalhar de forma colaborativa para a solução de determinados problemas algorítmicos. Dessa forma, os monitores tiveram a ideia de desenvolver uma atividade fora de sala (Figura 21): simular a travessia de missionários e canibais para o outro lado do rio.

Figura 21 - Atividade fora de sala de aula da dimensão Colaborar



Fonte: arquivo pessoal.

Esta atividade ocorreu em frente ao ICEN, em que foi usada a passarela existente para simbolizar um rio. A turma foi dividida em quatro grupos, e cada um dividiu-se entre missionários e canibais e recebeu as instruções dos monitores quanto ao desafio e suas particularidades. Após a explicação, os grupos resolveram a questão no papel e entregaram suas respostas aos monitores. Os monitores então liam os comandos para que os grupos pudessem executar a solução, dada por eles, utilizando a passarela (Figura 22).

Figura 22 - Execução do algoritmo realizado pelos grupos na atividade



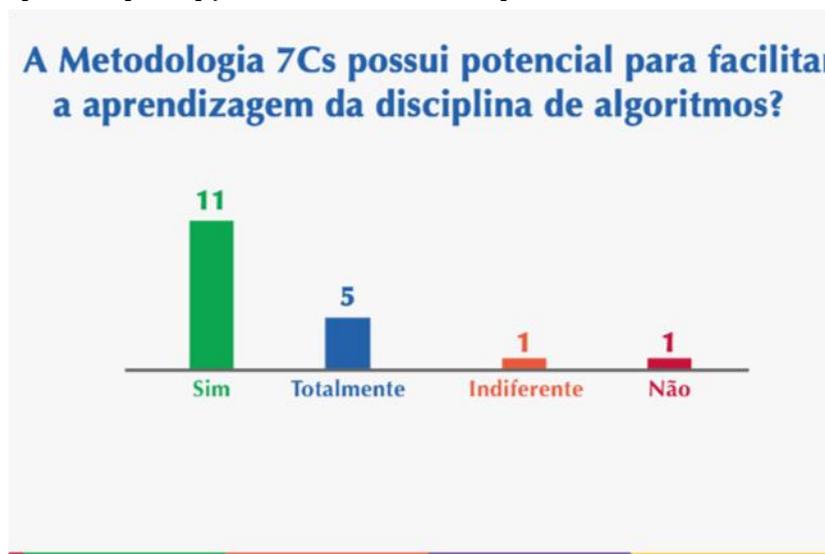
Fonte: arquivo pessoal.

O objetivo era que todos passassem para o outro lado sem que houvesse em momento algum um número maior de canibais em ambos os lados. Essa é uma questão clássica para exemplificar os passos para se chegar a uma solução de problemas e a conceituação de algoritmo, pelas sequências de passos realizados.

Alguns grupos tiveram dificuldade em concatenar as ideias e acabaram demorando mais para concretizar a atividade, porém ao final todos conseguiram finalizar o algoritmo com sucesso. A colaboração entre os alunos possibilitou a troca de conhecimentos e potencializou a compressão e o aprendizado dos conteúdos ministrados no curso.

Após a atividade, os discentes retornaram à sala de aula e participaram de uma pesquisa realizada no *Kahoot!* para verificar a opinião deles quanto à potencialidade da Metodologia 7Cs para a compreensão dos conteúdos ministrados durante o curso realizado (Figura 23).

Figura 23 - Pesquisa da percepção dos alunos sobre a potencialidade da Metodologia 7Cs

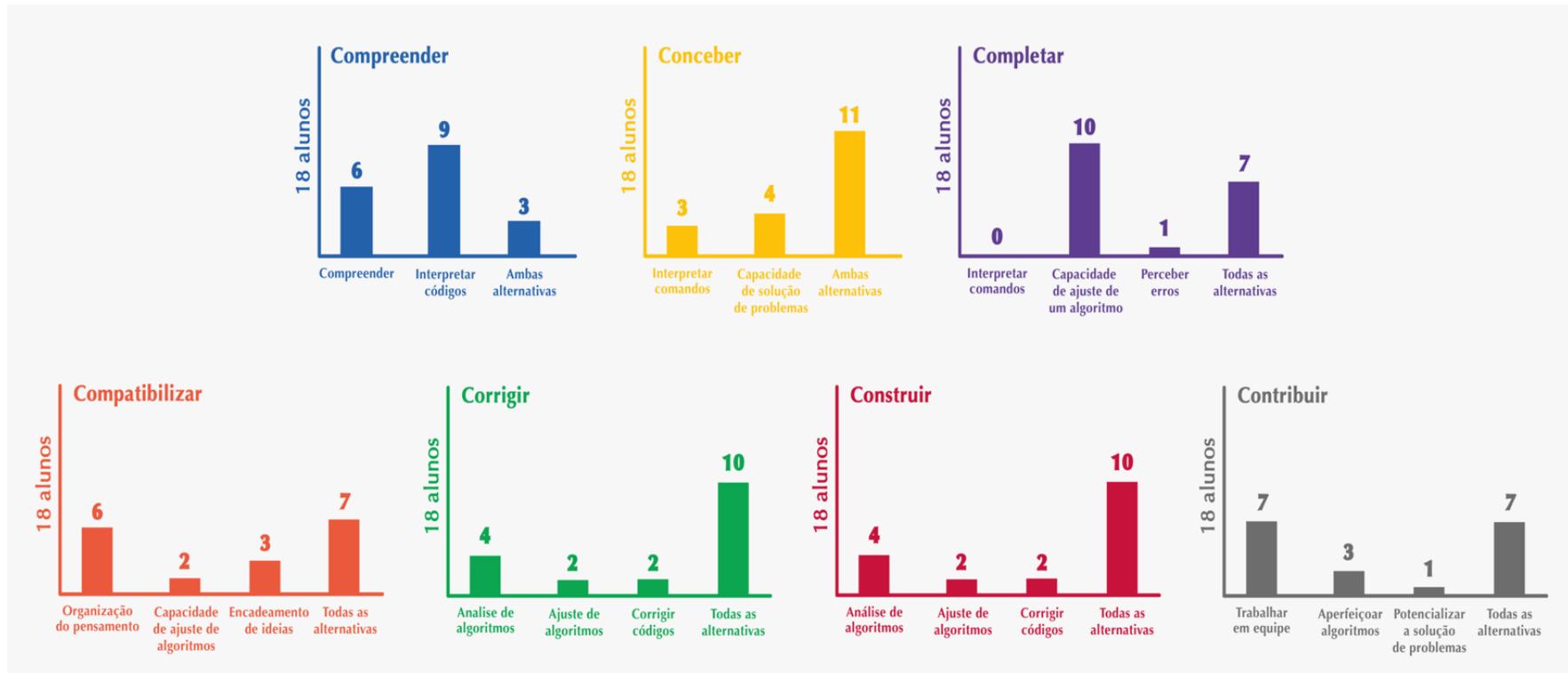


Fonte: Dados da pesquisa (2019).

Da pesquisa realizada, participaram 18 discentes, dos quais 11 afirmaram que “Sim” (ela pode ajudar de alguma forma na aprendizagem dos conteúdos da disciplina), cinco afirmaram “Totalmente” (com certeza é uma metodologia que de fato contribui para a aprendizagem dos conteúdos da metodologia), um afirmou ser “Indiferente” (não vai influenciar nem positivamente nem negativamente o processo de aprendizagem da disciplina) e um afirmou que “Não” (não possui potencial algum, podendo prejudicar no processo de aprendizagem da disciplina). A partir do resultado, inferimos que a Metodologia 7Cs foi bem avaliada, visto que a maioria considerou que ela possui potencial para contribuir no processo de aprendizagem dos alunos.

Conforme já citado no corpo do trabalho em tela, em cada dimensão presente na Metodologia 7Cs, foi elencado um conjunto de habilidades e competências que podem ser desenvolvidas durante sua aplicação. Assim, essa pesquisa também envolveu perguntar aos discentes que habilidades e competências entre as que elencamos eles consideraram ter desenvolvido durante o curso. A Figura 24 a seguir, mostra os resultados obtidos individualmente nas dimensões.

Figura 24 - Resultado sobre as habilidades e competências desenvolvidas na metodologia, na percepção dos alunos



Fonte: Elaborado pelo autor.

Após o teste piloto, produzimos um artigo para o Congresso da Sociedade Brasileira de Computação (CSBC2019)², um dos maiores eventos de Computação do Brasil. Nesse artigo publicado, relatamos a experiência vivenciada no PNA, na categoria de trabalhos de pesquisa em andamento, no Workshop sobre Educação em Computação. E para nossa imensa felicidade, nosso artigo foi premiado, recebendo uma menção honrosa pelo trabalho produzido.

A aplicação da Metodologia 7Cs no PNA foi um teste piloto com o objetivo de verificar a possibilidade de trabalhar os conteúdos ministrados na disciplina Algoritmos, bem como averiguar a possibilidade de outros professores aplicarem-na em sala de aula. Ao analisarmos nossas observações, acreditamos que os resultados foram positivos, pois os monitores não só conseguiram aplicar a metodologia respeitando sua concepção e sequência, como trouxeram outros elementos, ampliando a chance de tornar a aprendizagem significativa para os alunos. Além disso, foi possível obter os primeiros indícios de que a metodologia pode ser aplicada nos conteúdos da disciplina Algoritmos, sem prejudicar a forma como o professor ministra os conteúdos e, possivelmente, contribuindo para um melhor rendimento dos discentes.

5.2 APLICAÇÃO DA METODOLOGIA 7Cs EM DISCIPLINA DA GRADUAÇÃO

Após o teste piloto, sentimos a necessidade de realizar uma aplicação da metodologia em uma situação real de oferta da disciplina Algoritmos, o que ocorreu no primeiro semestre de 2019, com uma disciplina no CBCC, ofertada para os ingressos no curso. Essa disciplina trabalha conteúdos e estruturas de algoritmos que servem de base para as demais disciplinas do curso voltadas à programação.

Com o consentimento e orientação da professora da disciplina, desenvolvemos atividades para iniciar a aplicação da metodologia. Assim como no teste piloto realizado no Programa de Nivelamento de Algoritmos (PNA), primeiramente a professora ministrava as aulas teóricas sobre um determinado conteúdo didático, para posteriormente a metodologia ser empregada na parte prática da disciplina. Ficamos responsáveis pela produção dos materiais e condução em sala de aula. Criamos listas de exercícios e atividades que permitissem novos testes visando ao aprimoramento da proposta, para chegarmos finalmente na forma ideal que pretendíamos.

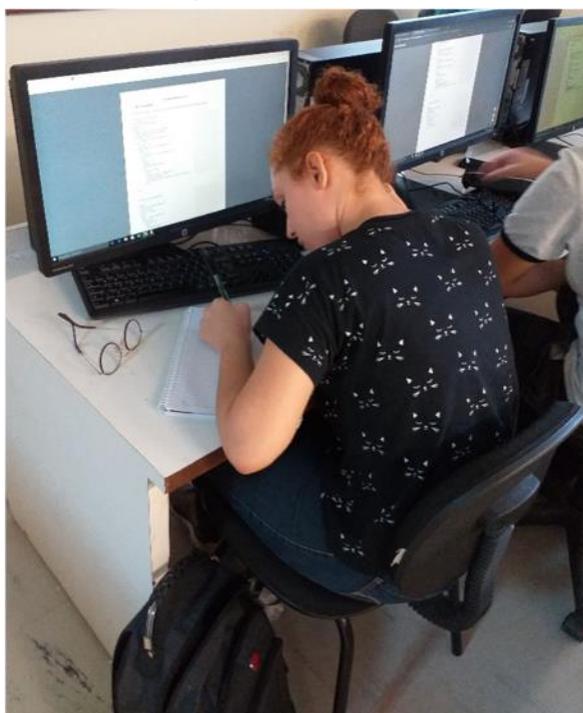
²Artigo publicado da Metodologia 7Cs: <https://sol.sbc.org.br/index.php/wei/article/view/6648/6544>

Embora grande parte dos alunos da disciplina tenha participado do PNA, eles experienciaram de forma diferente o uso da Metodologia 7Cs durante o semestre letivo, conforme relatamos a seguir.

5.2.1 Um teste da forma de aplicação: primeiras impressões

Para um primeiro teste, ao trabalharmos o conteúdo de estrutura de repetição, elaboramos duas listas de exercícios (Apêndice II) contemplando todas as dimensões previstas para as três fases, incluindo a dimensão Colaborar (retirada da versão final da proposta da Metodologia). Após a ministração do conteúdo, os alunos deveriam realizar todas as questões propostas até o encerramento da aula. Orientamos os discentes para seguir a ordem das questões, e assim respeitar a proposta da metodologia. Além disso, solicitamos que resolvessem as questões primeiramente no papel (Figura 25), para só então testá-las no computador.

Figura 25 - Resolução da atividade de maneira escrita

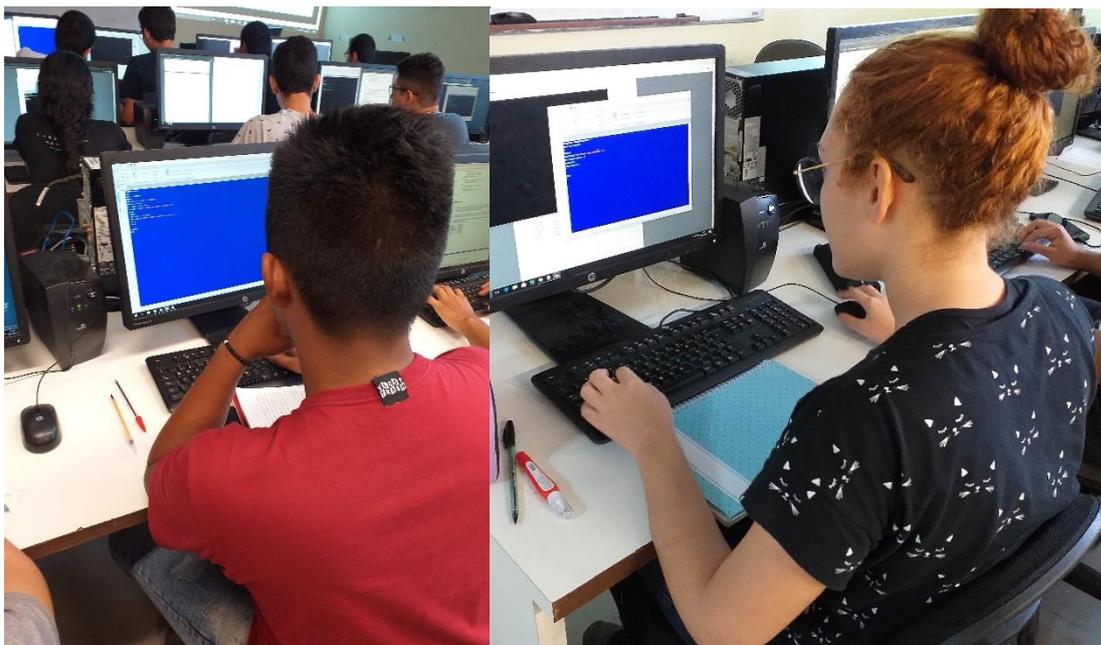


Fonte: arquivo pessoal.

A prática no papel pode contribuir para os discentes fixarem melhor os conteúdos ministrados. Além disso, nesse caso, apesar de em sala de aula os discentes terem acesso

ao computador, que permite trabalhar com tentativas de acertos e erros, em suas avaliações os códigos são construídos de forma escrita. Portanto, nesse sentido, a prática escrita foi muito importante no processo de aprendizagem dos alunos. Após a parte escrita, os alunos passaram suas soluções para o computador, como apresenta a Figura 26.

Figura 26 - Teste das respostas escritas nos computadores



Fonte: arquivo pessoal.

Durante a resolução da lista, os discentes puderam tirar dúvidas com o monitor, a professora e o autor deste trabalho de pesquisa (Figura 27). O monitor foi orientado sobre a Metodologia 7Cs para que melhor pudesse auxiliar os estudantes em conformidade com a proposta.

Figura 27 - Auxílio do monitor nas atividades em sala



Fonte: arquivo pessoal.

Ao observarmos a forma de solução e o comportamento dos discentes para resolver a lista de exercício, ocorreu algo que nos chamou atenção. Os discentes naturalmente interagiram entre si para discutir e compartilhar as soluções (Figura 28). Faz parte da concepção da Metodologia 7Cs estimular a criação de um ambiente de colaboração em que os discentes possam trocar conhecimentos e assim potencializar sua aprendizagem significativa (MOREIRA, 2011).

Figura 28 - Interação entre os alunos durante resolução da lista de exercício



Fonte: arquivo pessoal.

Inicialmente, percebemos que os alunos buscaram seguir as instruções de resolver a lista na ordem em que estavam as questões. Porém, identificamos que alguns começaram a pular questões, alegando que talvez eles conseguissem resolver algumas mais rapidamente do que outras. Essa situação poderia comprometer a forma como pretendíamos que eles construíssem seu processo de aprendizagem na disciplina, visto que não se trata de iniciar a resolução pelas mais fáceis, mas sim adquirir as competências necessárias ao aprendizado de maneira gradativa, conforme prevê cada dimensão da metodologia.

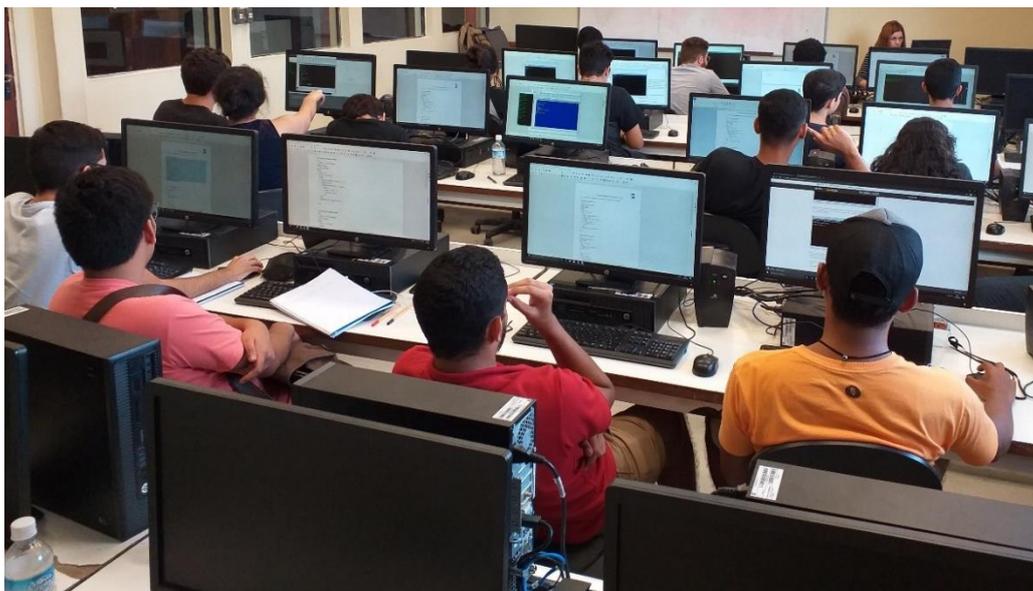
Percebemos a impossibilidade de os discentes resolverem as questões na ordem, mesmo que tivessem sido orientados para tal, quando tinham em mãos uma lista de exercícios contemplando todas as dimensões, pois sempre vão ter a curiosidade de ver e tentar responder às questões que acharem mais convenientes. Constatamos que nenhum deles conseguiu resolver todas as questões da lista, no tempo de aula da disciplina. Eram muitas informações para apenas duas horas de aula, principalmente por ser a primeira vez que eles praticavam exercícios com a proposta da Metodologia 7Cs.

Algumas dimensões da metodologia demandaram mais tempo e concentração dos discentes, e por isso não conseguiram realizar a lista por completo. Na dimensão Corrigir, por exemplo, verificamos que muitos deles demoram a compreender e corrigir os erros. Corrigir códigos de algoritmos construídos por outras pessoas não é uma tarefa fácil. Esse teste serviu para verificarmos a necessidade de produzir listas de exercícios cuja ordem lógica das fases da Metodologia 7Cs os discentes pudessem realmente seguir.

5.2.2 Novo teste

Decidimos realizar um novo teste, apenas contemplando a fase Descoberta com as dimensões Compreender e Conceber da metodologia, para verificar a viabilidade e forma como iríamos construir as próximas listas. A lista de exercício (Apêndice III) foi aplicada com o conteúdo referente à estrutura de dados homogênea de uma dimensão (vetor). A Figura 29 retrata os discentes trabalhando na lista.

Figura 29 - Lista de exercício disponíveis aos alunos

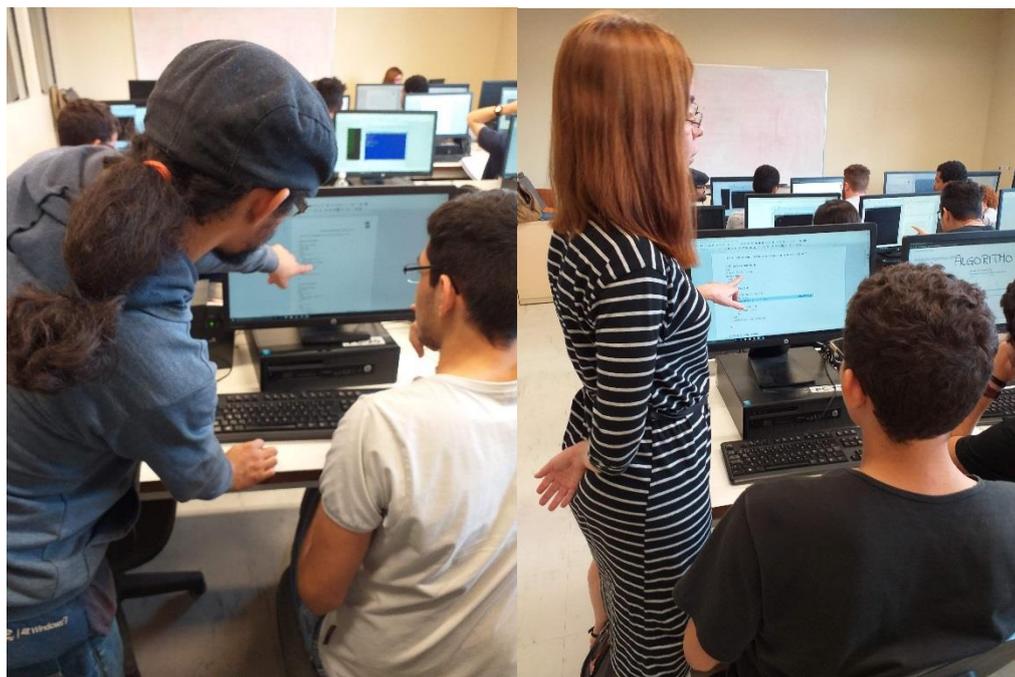


Fonte: arquivo pessoal.

A lista para o primeiro teste limitou a apenas um exercício para cada dimensão da metodologia, e mesmo assim o tempo da aula não foi suficiente para responder a todas. A nova lista, somente com a fase Descoberta, possibilitou aumentar o número de exercícios das dimensões correspondentes (Compreender e Conceber).

Consideramos positivo o aumento do número de questões, pois os discentes tiveram a oportunidade de praticar e desenvolver mais as habilidades e competências que cada dimensão proporciona. Solicitamos que primeiro fizessem a resolução dos problemas no papel e posteriormente no computador. Além disso, também podiam contar com o auxílio da professora da disciplina e do monitor sempre que necessário (Figura 30).

Figura 30 - Monitor e professora da disciplina auxiliando os alunos



Fonte: arquivo pessoal.

Neste teste chegamos a um ideal para aplicação final da Metodologia 7Cs na disciplina Algoritmos. Os discentes conseguiram concluir toda a lista e seguir a ordem proposta. Como já havíamos pensado confirmamos que de fato a metodologia precisa ser aplicada em fases, com tempo suficiente para adquirir de maneira adequada tudo o que ela proporciona.

5.2.3 Aplicação final da Metodologia 7Cs na disciplina Algoritmos

A partir do resultado do segundo teste produzimos três listas de exercícios voltadas à aprendizagem de Matrizes, cada qual com uma fase específica (Apêndice IV). As listas foram aplicadas em dias diferentes, o que possibilitou o aumento das questões para cada dimensão, bem como as listas serem resolvidas em sala de aula.

Os alunos se encontravam em nível bom de compreensão dos conteúdos, refletido nos resultados das avaliações. Não podemos afirmar que Metodologia 7Cs teve influência direta no avanço dos conteúdos. A maioria dos alunos, também já haviam participado do teste piloto da metodologia no PNA, e estavam passando por uma nova aplicação na própria disciplina. A seguir, apresentamos os dias de aplicação de cada fase da metodologia para o conteúdo referente à estrutura de dados homogênea de duas dimensões (matriz).

Primeiro dia de aplicação - Fase Descoberta

No primeiro dia de aplicação, os discentes receberam uma lista da fase Descoberta com as dimensões Compreender e Conceber. Assim como nos testes realizados, eles foram orientados a solucionar os problemas da lista primeiramente no papel e posteriormente no computador (Figura 31). Percebemos logo no início que os testes também serviram para familiarizar ainda mais os discentes com a metodologia, em que pouco foram as dúvidas sobre os exercícios e a proposta de cada uma das dimensões.

Figura 31 - Resolução das questões da lista no papel



Fonte: arquivo pessoal.

Ao pedirmos que os discentes resolvessem a lista de maneira escrita, buscamos que eles pudessem desenvolver as atividades sem maiores distrações, com nível maior de concentração e que não trabalhassem à base de tentativas e erros, como o computador permite, mas que verdadeiramente resolvessem os problemas, com base em seus conhecimentos.

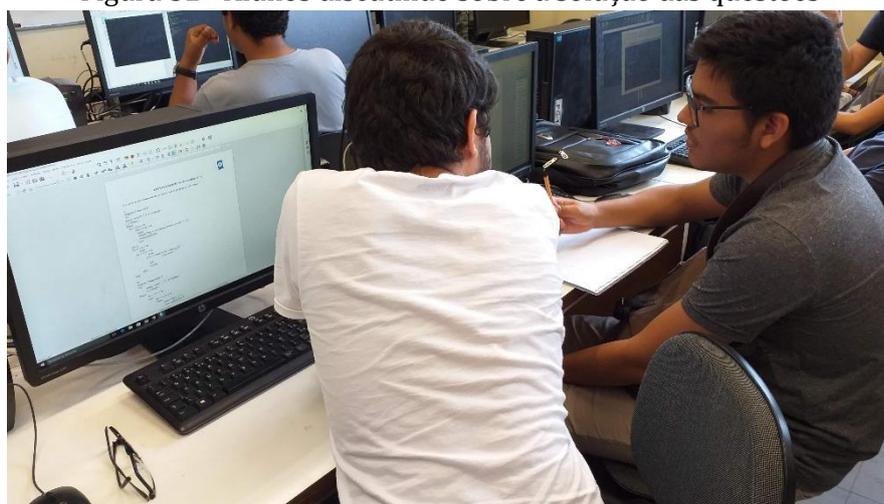
Na dimensão Compreender, os discentes tiveram que analisar e interpretar trechos de códigos de algoritmos, para descrever suas compreensões. Algo a se destacar é que, no primeiro teste de lista de exercício na disciplina, foi nítida a dificuldade de alguns discentes na dimensão Compreender para expressar de maneira escrita suas

compreensões acerca dos trechos de códigos de algoritmos. Mas, nesta aplicação, já percebemos uma certa facilidade e agilidade nas respostas.

O primeiro teste que realizamos serviu para que os discentes aprimorassem seus conhecimentos sobre a estrutura de algoritmos que estavam trabalhando (Repetição), bem como compreenderem melhor a proposta e funcionamento da Metodologia 7Cs. Tal feito resultou em uma melhor desenvoltura nas respostas e soluções dos problemas propostos nas listas de exercícios, com o uso da metodologia em matrizes.

Notamos, durante a resolução da lista de exercício, que os discentes começaram a interagir uns com os outros, para trocar informações de como estavam resolvendo as questões (Figura 32). Deixamos os discentes livres e incentivamos aqueles que haviam terminado suas atividades ou tivessem compreendido melhor o exercício a ajudar os colegas com mais dificuldades.

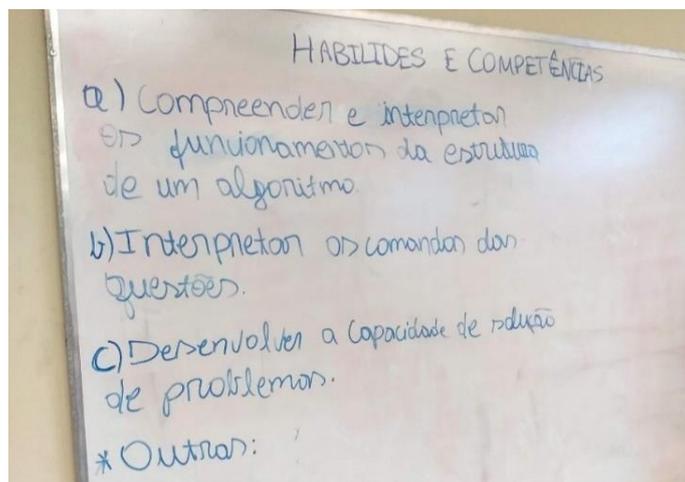
Figura 32 - Alunos discutindo sobre a solução das questões



Fonte: arquivo pessoal.

Conseguimos que os discentes terminassem a atividade em sala de aula, aproveitando melhor a prática de cada dimensão da fase Descoberta. No final da aula, colocamos no quadro as habilidades e competências que pretendíamos desenvolver, com as dimensões Compreender e Conceber (Figura 33). As habilidades e competências listadas foram: a) Compreender e interpretar os funcionamentos da estrutura de um algoritmo, b) Interpretar os comandos das questões e, c) Desenvolver a capacidade de solução de problemas.

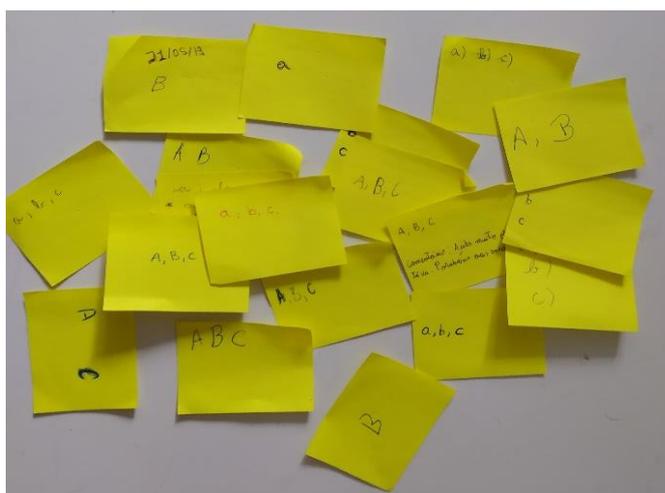
Figura 33 - Quadro magnético com as habilidades e competências da fase Descoberta



Fonte: arquivo pessoal.

As habilidades e competências das dimensões Compreender e Conceber foram mescladas e classificadas em alternativas a, b, e, c, como mostra a Figura 33. Os discentes tinham que destacar as alternativas que acreditavam ter desenvolvido em sala de aula na fase Descoberta. Para isso, distribuimos *post-its* para o destaque das alternativas (Figura 34). Além das habilidades e competências que destacamos, deixamos em aberto para que colocassem outras, que possivelmente poderiam ter desenvolvido.

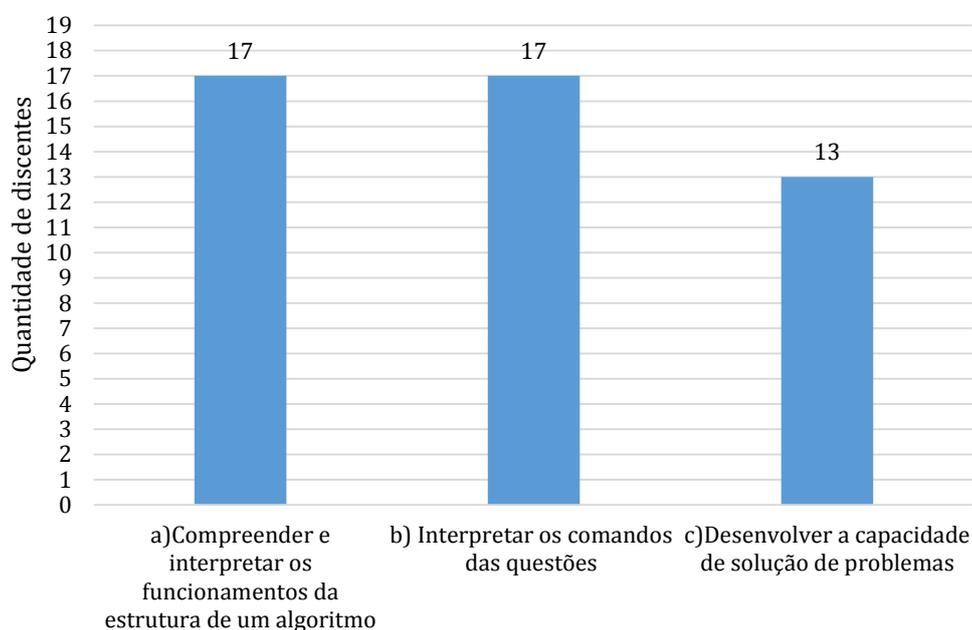
Figura 34 - Post-its usados para destaque das habilidades e competências



Fonte: arquivo pessoal.

Para melhor visualização das habilidades e competências escolhidas, construímos o Gráfico 1 a seguir, que demonstra a quantidade de alunos por alternativa.

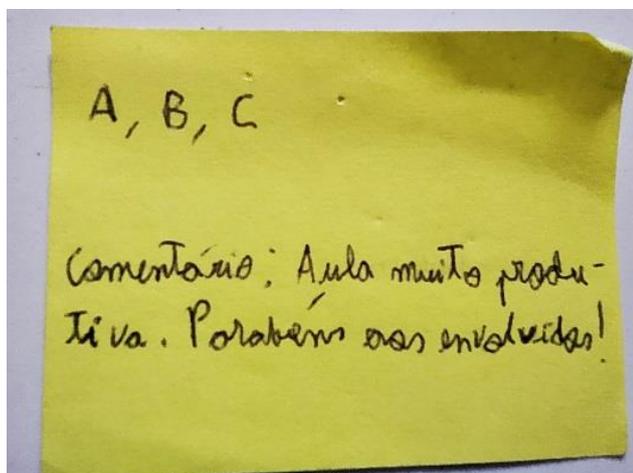
Gráfico 1 - Escolha pelos alunos das habilidades e competências da fase Descoberta



Fonte: Elaborado pelo autor.

Constatamos que a maioria dos discentes acredita, sim, ter desenvolvido as habilidades e competências elencadas na fase Descoberta. Nenhum deles conseguiu identificar ou quis opinar sobre outras habilidades e competências. Chamou a nossa atenção o comentário em um dos *post-its*: “Aula muito produtiva, parabéns aos envolvidos!” (Figura 35). Tal comentário denota que a Metodologia 7Cs possivelmente pode tornar o processo de aprendizagem mais produtivo e interessante aos discentes.

Figura 35 - Post-its com comentário de um dos alunos



Fonte: arquivo pessoal.

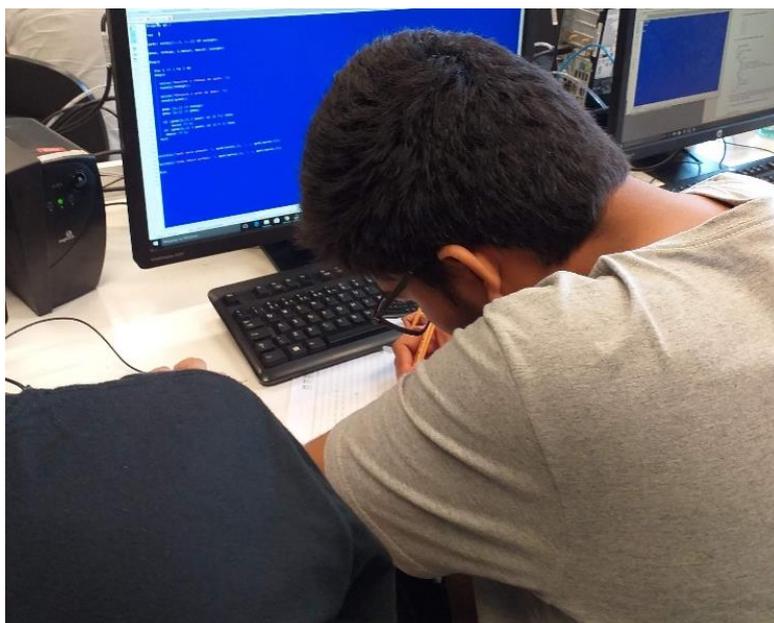
Quando elencamos as habilidades e competências da Metodologia 7Cs, pensamos no que seria necessário para ter um bom desempenho na disciplina Algoritmos. Por isso consideramos importante confirmar e ter um *feedback* dos alunos para saber se, na concepção deles, estávamos estimulando o desenvolvimento de tais habilidades e competências que elencamos para cada fase da Metodologia 7Cs.

Na fase Descoberta, os discentes primeiramente compreenderam a estrutura de algoritmos que estavam trabalhando (Matrizes) e conceberam as primeiras soluções de problemas mais simples, para adquirirem requisitos suficientes e conhecimentos preexistentes, que serviram como âncoras para a resolução da próxima lista com a segunda fase da metodologia, Aperfeiçoamento.

Segundo dia de aplicação – Fase Aperfeiçoamento

No segundo dia de aplicação, os discentes receberam uma lista da fase Aperfeiçoamento, com as dimensões Completar, Compatibilizar e Corrigir. Os discentes inicialmente resolveram a lista de exercícios no papel e posteriormente no computador (Figura 36). Com o tempo, os discentes foram se adaptando a essa maneira manual de chegar à solução dos problemas.

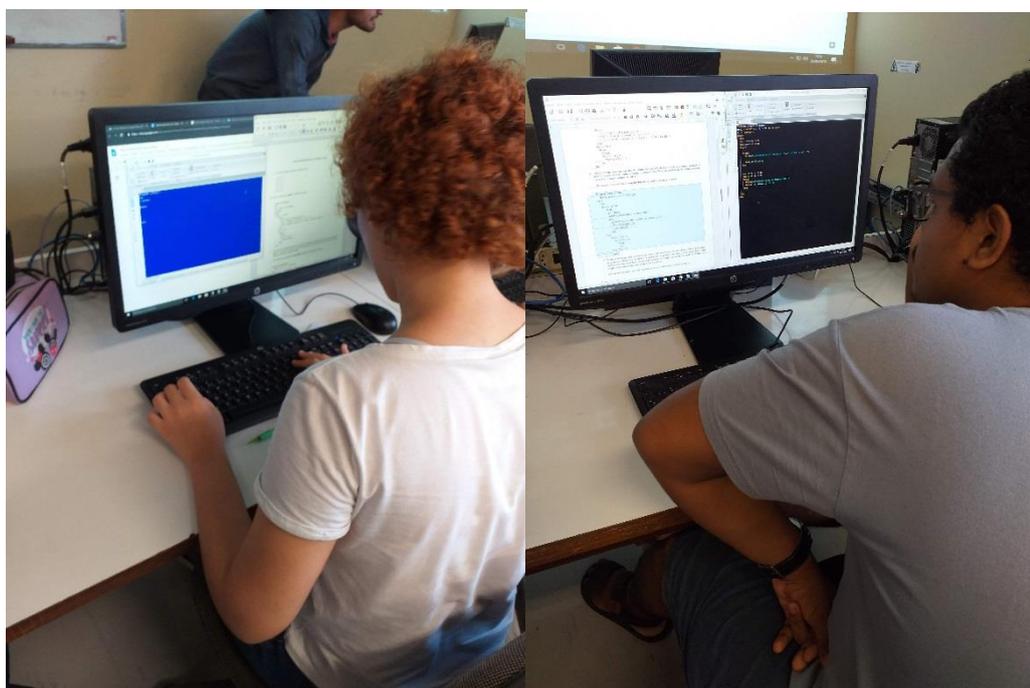
Figura 36 - Resolução da lista de exercício de maneira escrita



Fonte: arquivo pessoal.

Pedimos que, ao finalizarem cada uma das dimensões presentes na lista de exercícios, chamassem a professora, o monitor ou o autor desta pesquisa para que estes pudessem avaliar e validar cada resposta. Quando terminavam a atividade, os discentes, estavam livres para ajudar, se possível, os colegas que sentiam algum tipo de dificuldade. A Figura 37 a seguir mostra os discentes executando as soluções dos problemas no computador, após a resolução no papel.

Figura 37 - Execução das soluções no computador



Fonte: arquivo pessoal.

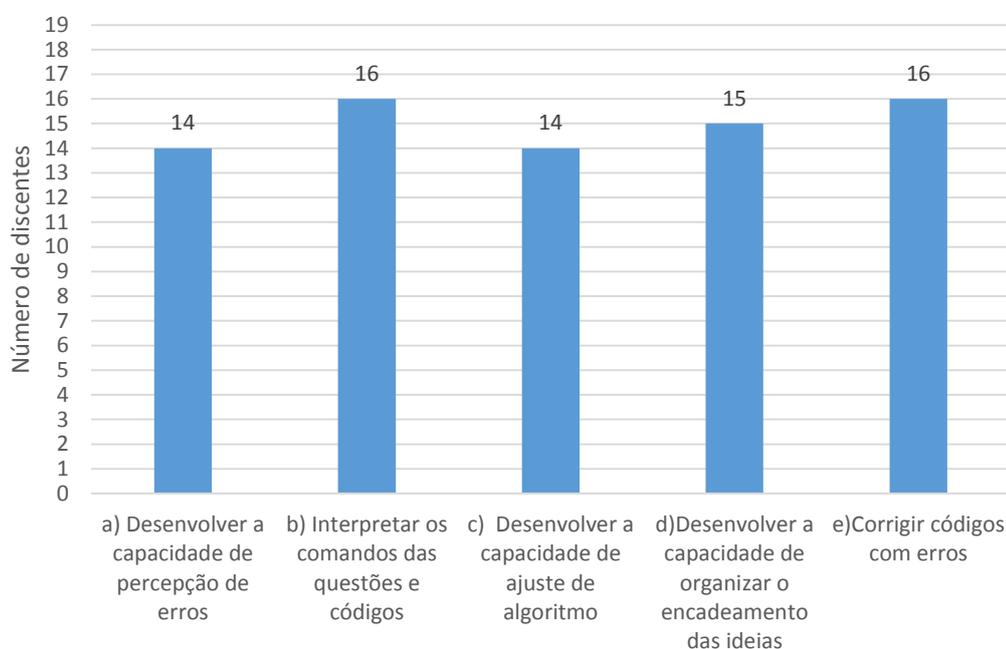
Percebemos, na fase Aperfeiçoamento, a demora de alguns dos discentes para chegar à solução dos problemas. As dimensões dessa fase trabalham com algoritmos prontos, fazendo com que os discentes pudessem verificar sua capacidade e o seu potencial para conseguir completar, compatibilizar e corrigir algoritmos, mesmo que não fossem construídos por eles.

No final da aula, colocamos no quadro as habilidades e competências mescladas, das dimensões Completar, Compatibilizar e Corrigir. As habilidades e competências listadas foram: a) interpretar os comandos das questões e códigos; b) interpretar os comandos das questões; c) desenvolver a capacidade de percepção de erros; d) desenvolver a capacidade de ajuste de algoritmo; e) desenvolver a capacidade de

organizar o encadeamento das ideias; f) Corrigir códigos com erros; e g) desenvolver a capacidade de análise de algoritmos.

Pedimos que os discentes destacassem as alternativas nos *post-its* que disponibilizamos. Contamos com a participação de 19 discentes. Ao analisarmos os *post-its*, verificamos uma diversidade de visões sobre as habilidades e competências que os discentes possivelmente teriam desenvolvido em sala de aula. A seguir, o Gráfico 2 mostra a quantidade de discentes por habilidade e competência.

Gráfico 2 - Escolha pelos alunos das habilidades e competências da fase Aperfeiçoamento



Fonte: Elaborado pelo autor.

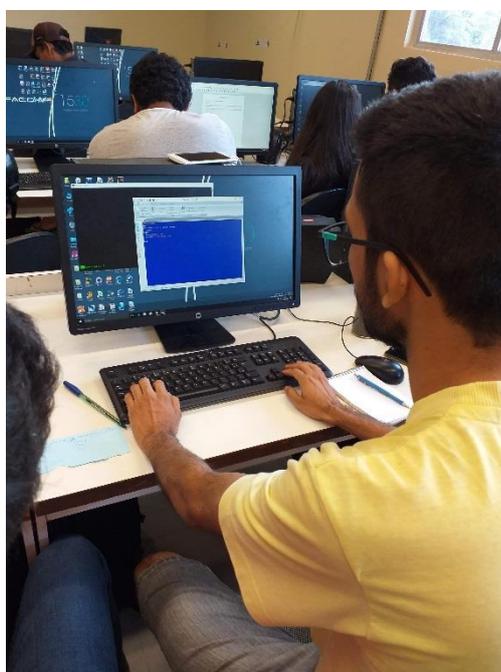
Ao considerarmos o resultado, percebemos que as habilidades e competências que elencamos para a fase Aperfeiçoamento não estão distantes da visão dos discentes, pois a maioria deles acredita tê-las desenvolvido durante a execução dessa fase.

Quando construímos um algoritmo, é necessário termos sensibilidade e coerência sobre os elementos a serem inseridos no código, para que a solução do problema seja realizada com eficiência, sem erros em sua estrutura. Portanto, a fase Aperfeiçoamento teve como importância a percepção dos discentes sobre o valor da construção de algoritmos com todos os elementos necessários, identificando possíveis erros no encadeamento nas linhas de código e no próprio algoritmo.

Terceiro dia de aplicação – Fase Consolidação

No terceiro dia de aplicação, os alunos receberam uma lista de exercícios referente à fase Consolidação contemplando a dimensão Construir. Em seguida, realizamos uma atividade envolvendo a dimensão Criar. A lista de exercícios contava com questões de nível de dificuldade maior, e os alunos ficaram livres para resolvê-las no papel ou diretamente no computador (Figura 38).

Figura 38 - Resolução da lista da dimensão Construir



Fonte: arquivo pessoal.

Durante a atividade, percebemos os discentes mais confiantes e cuidadosos na construção da solução dos problemas. Mesmo com um grau de complexidade maior, verificamos que não houve dificuldades para resolverem os problemas propostos. Esse foi um fato positivo, visto que a dimensão Construir é parte natural da disciplina, porém é comum os discentes sentirem dificuldades por se tratar de problemas mais complexos.

Ao finalizarem a lista de exercícios, passamos para dimensão Criar. Para isso, propomos uma atividade que intitulamos “É Hora de Criar”, realizada conforme as etapas descritas no Quadro 17.

Quadro 17 – Etapas para dinâmica da dimensão Criar

ETAPA	DESCRIÇÃO
Etapa 1	Dividimos os alunos em quatro grupos com base no desempenho das avaliações e atividades realizadas, procurando mesclar os que considerávamos ter mais desenvoltura na disciplina com os que tinham menos. A Figura 39 apresenta a forma como os discentes ficaram distribuídos.
Etapa 2	Explicamos a atividade com as seguintes orientações: a) os grupos devem criar um comando de questão, em 30 minutos; b) o comando de questão deve ser feito em um papel separado (Figura 40); c) devem levar em consideração a viabilidade de solução do problema e o grau de complexidade aceitável, ao considerar tudo o que já foi visto; e d) a construção do comando deve ter a participação de todos os integrantes do grupo.
Etapa 3	Os grupos deveriam desenvolver e finalizar comandos de questões e participar de um sorteio da seguinte forma: a) os grupos deveriam entregar os comandos das questões em um papel separado; b) os grupos foram identificados pelos números de 1 a 4 para a realização de um sorteio; e c) cada grupo sorteou um número diferente do seu; caso contrário, o devolveria para ser sorteado novamente.
Etapa 4	Realizamos a última parte da atividade da seguinte forma: a) o número sorteado correspondia ao número do comando da questão feita por um grupo diferente do seu, para que solucionassem o problema proposto pelos colegas; e b) todos os integrantes do grupo deveriam participar de forma colaborativa da atividade, na busca de solucionar o problema.

Fonte: Elaborado pelo autor.

Figura 39 - Formação dos grupos para o desenvolvimento da atividade



Fonte: arquivo pessoal.

Figura 40 - Grupo construindo um comando de questão no papel



Fonte: arquivo pessoal.

Verificamos que os discentes estavam engajados e dispostos a realizar o desafio proposto. A possibilidade de eles mesmos criarem seu próprio comando de questão com uma situação problema foi essencial para que eles pudessem, a partir de seus conhecimentos, verificar suas capacidades, não só para solucionar problemas já construídos, mas também criá-los considerando tudo o que haviam aprendido até aquele momento.

A dimensão Criar proporcionou um ambiente colaborativo e de estímulo ao processo criativo, o qual potencializou a aprendizagem dos discentes por meio da interação e compartilhamento de conhecimentos, tornando o processo mais significativo. Aprender de maneira colaborativa é proporcionar aos discentes a possibilidade de participar mutuamente na construção de algo, utilizando e trocando conhecimentos. A Figura 41 retrata a interação ocorrida entre os discentes durante a atividade.

Figura 41 - Interação entre os integrantes do grupo para a solução da atividade



Fonte: arquivo pessoal.

Nesse processo, contamos com a ajuda do monitor para supervisionar e auxiliar nas dúvidas dos discentes (Figura 42). A supervisão foi necessária para verificar a participação de todos os integrantes dos grupos na construção das soluções dos problemas, bem como para que os discentes pudessem focar na atividade sem maiores distrações. Para isso, por exemplo, pedimos que fosse utilizado apenas um computador por grupo para realizar a atividade.

Figura 42 - Monitor supervisionado e auxiliando os alunos



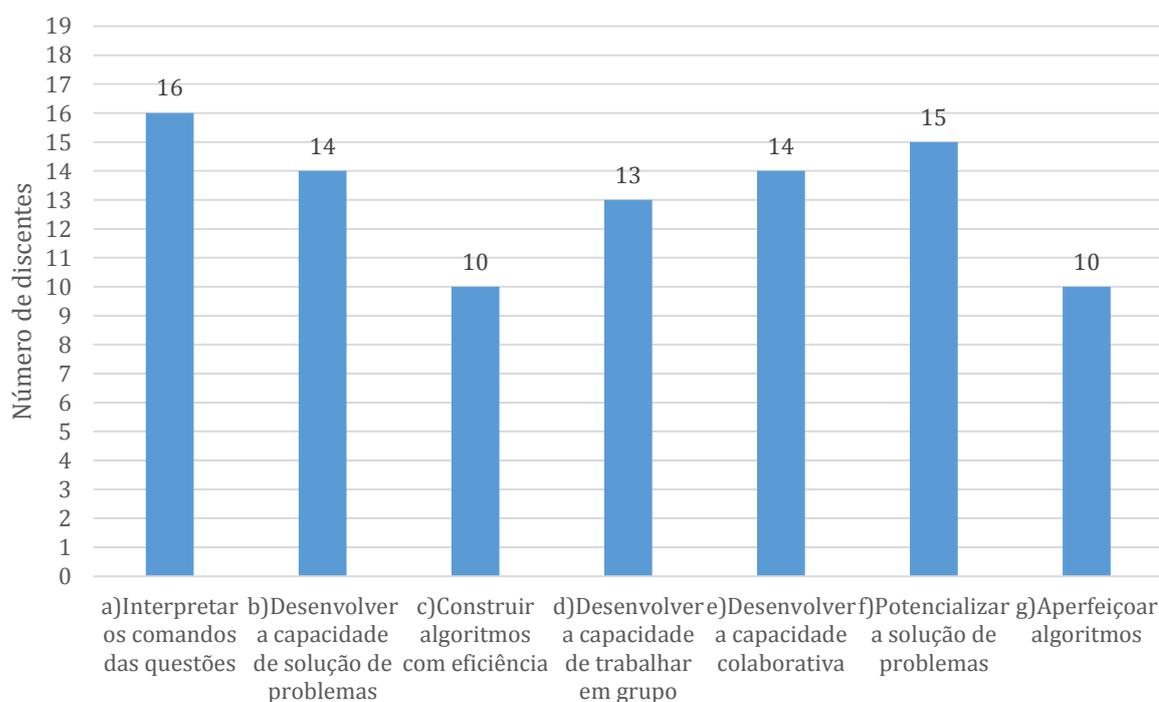
Fonte: arquivo pessoal.

Para avaliar as respostas dos grupos, pedimos que, ao resolverem a atividade, nos chamassem para que validássemos a resposta. Nesse processo, solicitamos que um representante do grupo que criou o comando da questão nos ajudasse a avaliar se a solução estava de acordo com o que fora solicitado. Foi importante permitir que discentes participassem da avaliação para que pudessem se sentir parte do processo de ensino.

Ao final da aula, assim como nas demais fases da metodologia, redigimos no quadro as habilidades e competências que pretendíamos desenvolver com as dimensões Construir e Criar. As habilidades e competências listadas foram: a) interpretar os comandos das questões; b) desenvolver a capacidade de solução de problemas; c) construir algoritmos com eficiência; d) desenvolver a capacidade de trabalhar em grupo; e) desenvolver a capacidade colaborativa; f) potencializar a solução de problemas; e g) aperfeiçoar algoritmos.

Solicitamos mais uma vez que os discentes destacassem nos *post-its* que disponibilizamos as alternativas que consideravam ter desenvolvido em sala e indicar outras se achassem necessário. O Gráfico 3 apresenta o número de discentes por habilidade e competência.

Gráfico 3 - Escolha pelos alunos das habilidades e competências da fase Consolidação



Fonte: Elaborado pelo autor.

Assim como nas demais fases, obtivemos resultado considerado positivo, visto que 50% ou mais dos discentes consideraram ter desenvolvido as habilidades na fase Consolidação. Na concepção final da Metodologia 7Cs, realizamos acréscimo do estímulo ao processo criativo, como uma das habilidades e competências da dimensão Criar.

Nesta fase, percebemos uma evolução e facilidade da maioria dos discentes em solucionar problemas considerados complexos. Não podemos de fato afirmar que a Metodologia 7Cs teve influência sobre a desenvoltura dos discentes, mas estes apresentaram uma melhora significativa do desempenho na disciplina Algoritmos. A fase de Consolidação foi importante para que os discentes pudessem solidificar os conhecimentos adquiridos durante o conteúdo ministrado e verificar suas potencialidades e evolução no decorrer da disciplina Algoritmos.

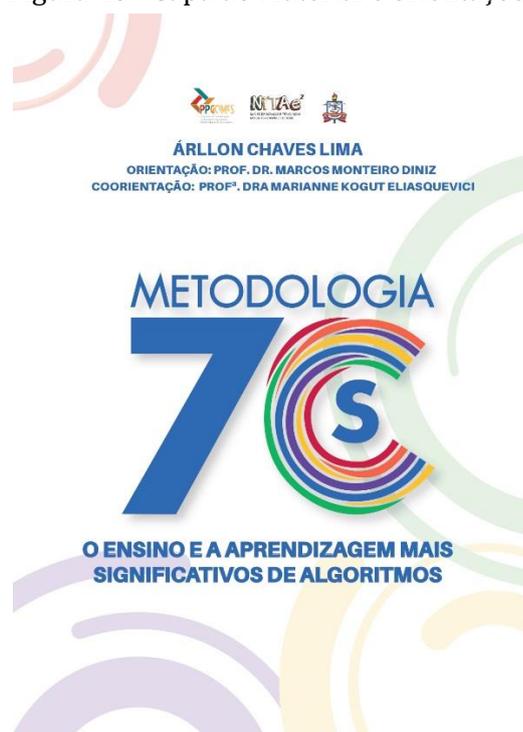
6 MATERIAL DE ORIENTAÇÕES DA METODOLOGIA 7Cs

Neste capítulo discriminamos os objetivos do material de orientações, criado para permitir a outros professores empregarem a Metodologia 7Cs em seus contextos de aula. Assim, descrevemos a motivação para sua concepção, os seus requisitos, a organização das seções, o projeto gráfico escolhido e a forma como pretendemos divulgar e ampliar a Metodologia 7Cs.

6. 1 POR QUE CRIAR UM MATERIAL DE ORIENTAÇÕES?

O produto desta dissertação, conforme já citado, é o processo de concepção e construção da Metodologia 7Cs. Porém, no Mestrado Profissional é necessário “gerar um produto educacional que possa ser disseminado, analisado e utilizado por outros professores” (MOREIRA; NARDI, 2009, p.4). Por esse motivo, para que professores de disciplinas relacionadas à programação possam ter acesso à metodologia, elaboramos um material de orientações, intitulado “METODOLOGIA 7Cs: O Ensino e a Aprendizagem mais significativos de algoritmos”, para explicar como esta pode ser empregada em situação real de sala de aula (Figura 43).

Figura 43 - Capa do Material e orientações



Fonte: Elaborado pelo autor.

Por meio do uso do material de orientações, teremos a oportunidade de verificar a abrangência da metodologia, seu alcance, seu potencial de replicabilidade e adequação, além de identificar se, de fato, a metodologia atinge o objetivo esperado, em diferentes realidades. Vai ser possível, também, obter o *feedback* de professores que fizerem uso do material, para indicar os resultados obtidos e sugestões de aprimoramento da metodologia.

6.2 REQUISITOS PARA A CONCEPÇÃO DO MATERIAL

Inspirados em materiais desenvolvidos para serem utilizados no contexto da Educação a Distância (EaD), buscamos produzir um texto didático com a intenção de ensinar ao leitor de forma clara, como utilizar a Metodologias 7Cs. Segundo Petri (2010, p.21) a “função do material didático, para quem o elabora para cursos a distância, é **ensinar**, mas tendo como objetivo central a **aprendizagem** de quem estuda em casa, no trabalho, sem a presença física do professor.” Portanto, tínhamos como objetivo criar um material, autossuficiente, para possibilitar ao nosso público (professores de graduação) aprender, compreender e ter caminhos para executar a proposta. Assim foram definidos dois grupos de requisitos bases para a concepção do material de orientações denominados de funcionais e estruturais.

Conforme Petri (2010), um texto didático para educação a distância deve desempenhar algumas funções importantes ao processo de aprendizagem do conteúdo. Dentre as funções que este autor apresenta, destacamos, no Quadro 18, as que adequamos como **requisitos funcionais**, em relação ao pretendido com a concepção do material de orientações da Metodologia 7Cs.

Quadro 18 -- Requisitos funcionais do material de orientações

FUNÇÕES DO TEXTO DIDÁTICO (PETRI, 2010) / REQUISITOS	MATERIAL DE ORIENTAÇÕES
Favorecer o desenvolvimento de [...] atitudes	Queremos incentivar os professores a terem atitudes que visem minimizar as dificuldades dos alunos em disciplinas introdutórias de programação.
Integrar a teoria com a prática	Apresentamos duas listas de exercícios com propostas de atividades de aprendizagem, como forma de os professores integrem a teoria com a prática.
Provocar questionamento reconstutivo e a capacidade de estudo autônomo	Buscamos estruturar o texto de maneira clara, para que os professores pudessem compreender como usar a metodologia para conseguirem aplicá-la de forma autônoma.
Propiciar leitura agradável e compreensiva	Buscamos deixar o texto do material simples e direto, mas com todo rigor científico necessário, para facilitar a compreensão dos conteúdos abordados.
Manter diálogo com o estudante	Utilizamos na concepção do material uma linguagem dialógica, característica de materiais para EaD, para ter uma maior proximidade com o leitor, dialogando e chamando os professores a refletir sobre suas vivências, relacionadas com os conteúdos apresentados.
Motivar	Buscamos motivar os professores tanto por meio do cuidado em encadear as informações de forma objetiva e ilustrativa, quanto ao mostrar que por meio do uso da Metodologia 7Cs eles podem fazer a diferença para seus alunos.

Fonte: Elaborado pelo autor.

Petri (2010) também destaca elementos estruturais característicos de um texto didático, que nos serviram de inspiração para elencar os requisitos estruturais para composição textual e visual do material de orientações. Dentre os elementos citados pelo autor, destacamos os seguintes, apresentados e descritos no Quadro 19: conteúdo; ilustrações; tom e estilo; e orientações.

Quadro 19 - Requisitos estruturais do material de orientações

CARACTERÍSTICAS TEXTO DIDÁTICO (PETRI, 2010) / REQUISITOS		MATERIAL DE ORIENTAÇÕES
ELEMENTOS	DESCRIÇÃO	
Conteúdo	Deve levar em consideração as demandas e necessidades de determinados grupos.	Buscamos desenvolver uma proposta de metodologia ensino e aprendizagem com base no contexto vivenciado por muitos alunos, em dificuldades na compreensão dos conteúdos introdutórios de programação.
Ilustração	Imagens, fotos, diagramas, tabelas, entre outras ilustrações utilizadas ativamente, como recurso didático para melhor explicar determinado conteúdo.	Buscamos tornar o material mais ilustrativo, ao visarmos facilitar a compreensão e entendimento da Metodologia 7Cs.
Tom e estilo	“Faz uso de linguagem menos formal e mais coloquial - sem perder a precisão científica e técnica -, procurando envolver o estudante” (PETRI, 2010, p.22).	Buscamos constituir uma linguagem simples e direta, mas com objetivo torna o texto mais limpo, apenas com as informações essenciais.
Orientações	“São [...] sugeridos procedimentos ou passos a serem seguidos para obter sucesso em determinada tarefa [...] (PETRI, 2010, p.23).	Buscamos realizar orientações no material, em relação ao uso da Metodologia 7Cs, inserindo exemplos e sugestões, para auxiliar os professores na aplicação prática da metodologia.

Fonte: Elaborado pelo autor.

Os requisitos refletem o cuidado com a concepção e desenvolvimento do o material de orientações para alcançar as necessidades e objetivos almejados, com o intuito de possibilitar o fácil manuseio do material para aplicação da Metodologia 7Cs e o uso dos recursos que ele disponibiliza aos professores (como o jogo M7C e as listas de exercícios).

6.3 ORGANIZAÇÃO DO MATERIAL

O conteúdo disponibilizado no material de orientações foi organizado e sequenciado levando em consideração o seu propósito e a necessidade de os professores, por meio da leitura do material de orientações, conseguirem compreender as motivações, os objetivos, a base teórica e a importância da Metodologia 7Cs para o ensino e aprendizagem de algoritmos. Apresentamos no Quadro 20, as seções do material de orientações com suas respectivas descrições.

Quadro 20 - Descrições das seções que compõe o material de orientações

SEÇÃO	DESCRIÇÃO
Convite aos(as) professores(as)	Estabelecimento de contato inicial com os professores, com o objetivo de seduzi-los e mostrar a importância do material.
Nova metodologia, novas possibilidades	Apresentação do contexto sobre as dificuldades vivenciadas pelos alunos em disciplinas introdutórias de programação. Buscamos mostrar, por meio de uma experiência real, que nós professores um dia fomos alunos e passamos por dificuldades, semelhantes ou iguais, as que muitos passam ainda hoje.
Um mergulho na Metodologia 7Cs	Apresentação das particularidades da Metodologia 7Cs, a base teórica que a fundamenta (Aprendizagem Significativa de Ausubel (2000) e o motivo de estruturá-la em três fases.
A Metodologia 7Cs em ação	Exemplificação de uma possível forma de aplicação da Metodologia 7Cs em sala de aula, de forma a facilitar a compreensão pelos professores, incluindo o uso de um jogo.
Sugestão de listas	Sugestão de duas listas de exercícios (estrutura de repetição e matrizes), contemplando as fases e dimensões da metodologia, para inspirar os professores na elaboração de novas listas.
Preparados(as)?	Finalização do material, com o incentivo aos professores em utilizar a Metodologia 7Cs em sala de aula e participar de um grupo no Facebook para compartilhamento da experiência.
Referências	Apresentação das Referências bibliográficas utilizadas na composição textual do material.
Gabarito do jogo M7Cs	Gabarito do jogo M7C completo para impressão, com objetivo de facilitar o uso pelo professor.

Fonte: Elaborado pelo autor.

6. 4 PROJETO GRÁFICO

O projeto gráfico do material de orientações, constituiu os elementos e as características dos conteúdos abordados. Na concepção de um material de didático é essencial termos a noção de que de além do seu conteúdo verbal e ilustrativo, existem vários elementos, que constituem o *layout*, usados como estratégia, para promover a motivação e organização, em conjunto ao processo de aprendizagem (WEBER E OLIVEIRA, 2016). Buscamos deixar o material com linguagem simples e de uma maneira que pudéssemos dialogar com o professor (Figura 44), além tentar deixar o material visualmente agradável, para motivar a leitura e compreensão da proposta da metodologia.

Figura 44 - Dialogando com os professores

Professores(as), vocês sabem o quanto a disciplina Algoritmos é importante, principalmente em cursos na área de TI, pois é base para outras disciplinas de programação, ainda mais complexas. Para Barcelos *et al.* (2009), o estudo de algoritmos é uma área fundamental no campo da Computação, pois o conhecimento e a compreensão desse conteúdo permitirá a aprendizagem nas linguagens de programação, que em sua essência possuem estratégias de solução de problemas.

A disciplina exige habilidades e competências, tais como raciocínio lógico, leitura e interpretação, capacidade de abstração, domínio de conceitos matemáticos, resolução de problemas, entre outras (RAABE e SILVA, 2005; GOMES e MENDES, 2007; PIVA JÚNIOR e FREITAS, 2010), de maneira integrada, o que não é percebido em grande parte dos alunos, desde a educação básica, interferindo diretamente na aprendizagem e na compreensão dos conteúdos ministrados por vocês.

Confesso a vocês que na graduação em Licenciatura em Computação, realizada na Universidade Federal Rural da Amazônia (UFRA), enfrentei grandes desafios na disciplina Algoritmos. Era algo totalmente novo, que não fazia parte dos conhecimentos que havia construído até ali. Sentia-me muitas vezes frustrado e desmotivado a cada erro cometido na solução dos problemas que me eram postos, o que piorava quando o grau de complexidade dos problemas aumentava. Assim como eu, grande parte dos alunos egressos do ensino médio regular também não possui base técnica de lógica ou algoritmos, o que reflete no aumento da dificuldade de aprendizado na disciplina (VIEIRA; LIMA JUNIOR; VIEIRA, 2015).

Então, imaginem: se seus alunos têm dificuldade em assimilar os conteúdos ministrados e por isso não conseguem obter um bom rendimento, provavelmente terão dificuldade ao longo do curso, gerando um sentimento de desmotivação e frustração, que muitas vezes provoca altos índices de reprovação e evasão. Conforme Giraffa e Mora (2013), a evasão em disciplinas dos níveis iniciais dos cursos de Computação não é um tema novo, é algo recorrente e que vem sendo muito estudado, tanto em nível nacional quanto internacional.

Acreditamos que vocês, quando alunos, tenham passado por situação semelhante. Deve ser incômodo saber que ministram uma



6

Fonte: Elaborado pelo autor.

A fonte é um elemento importante que possui a função de tornar visíveis as informações que pretendemos transmitir e deve complementar o conteúdo, mas não competir com ele (WEBER; OLIVEIRA, 2016). Por isso, buscamos escolher uma fonte agradável e confortável ao leitor, para fazer sentido em relação ao que queremos apresentar. O pitamos por usar uma fonte simples e em determinados momentos, usamos diferentes fontes para que pudesse ficar mais claro visualmente algumas informações, como por exemplo, os códigos de algoritmos presentes no material (Figura 45).

Figura 45 - Códigos de algoritmos

Observem que:

- É importante que os algoritmos ou trechos destes sejam escolhidos cuidadosamente, pois é o primeiro contato dos alunos com a estrutura trabalhada;
- É interessante que os alunos realizem as atividades no papel, para que a aprendizagem seja mais construtiva à medida que os eles adquirem um maior nível de concentração.

Para ficar mais compreensível, deixamos aqui um exemplo de exercício:

- A partir dos trechos de algoritmos a seguir, descreva, de acordo com sua compreensão, o que será executado:

a) Var

```

Num: integer;
Begin
  Writeln ('digite um número');
  Readln (num);
  Writeln ('o antecessor de ', num, ' eh: ', num-1);
End.

```

b) num:= 10;

```

For cont:= -1 to 3 do
Begin
  Writeln (num);
  Num:= num - cont;
End;

```

Obs: vejam que usamos neste exemplo duas estruturas diferentes (a - estrutura sequencial, b - estrutura de repetição) apenas como demonstração, mas na prática devem ser aplicadas no momento em que for iniciado o ensino de suas respectivas estruturas.

Conceber (C2)

Conceber é o primeiro passo para se concretizar algo, que ainda pode ser amadurecido e trabalhado.

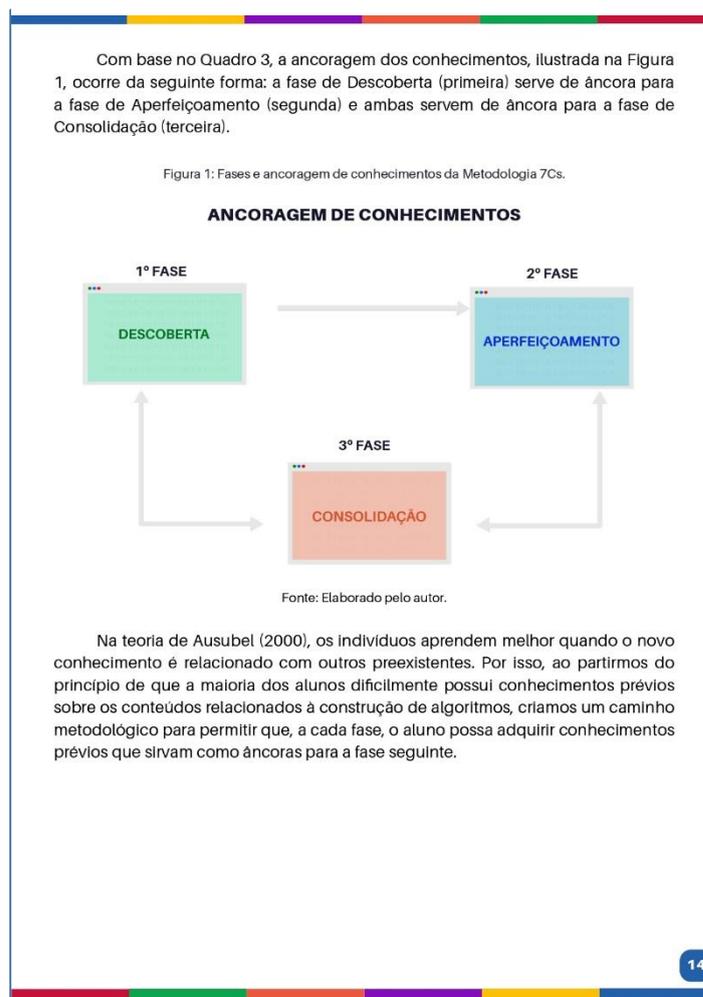
É o momento em que os alunos **concebem** seus primeiros algoritmos simples e, assim, demonstram a sua compreensão sobre a estrutura de algoritmos trabalhada.

19

Fonte: Elaborado pelo autor.

As imagens também se constituem como um elemento importante, pois segundo Weber e Oliveira (2016, p.) “são emocionais, despertam curiosidade e introduzem e/ou complementam a informação ao leitor, por isso possuem papel importante em materiais didáticos”. Nesse sentido, buscamos construir dentro do texto ilustrações, para estimular visualmente a leitura e compreensão pelos professores (Figura 46).

Figura 46 - Ilustração das fases da Metodologia 7Cs



Fonte: Elaborado pelo autor.

É importante estar atento na captura do leitor sobre o material, para que mesmo antes de chegar ao texto, ele possa ser motivado pelo uso de imagens e títulos sugestivos (Weber e Oliveira, 2016). Na concepção do material, utilizamos títulos e subtítulos de maneira sugestiva para seduzir os professores e estimular a curiosidade sobre as informações abordadas (Figura 47).

Figura 47 - Sumário do material de orientações

SUMÁRIO	
I NOVA METODOLOGIA, NOVAS POSSIBILIDADES.....	5
II UM MERGULHO NA METODOLOGIA 7CS.....	8
2.1 A teoria inspiradora: aprendizagem significativa e ancoragem de conceitos.....	9
2.2 Descrevendo a Metodologia 7Cs	10
III A METODOLOGIA 7CS EM AÇÃO.....	15
3.1 Pontapé Inicial.....	17
3.2 Explorando as fases.....	18
3.2.1 Descoberta.....	18
3.2.2 Aperfeiçoamento.....	20
3.2.3 Consolidação	25
IV SUGESTÃO DE LISTAS	29
4.1 Lista 1: estrutura de repetição.....	30
4.2 Lista 2: matrizes.....	33
MÃOS À OBRA.....	37
REFERÊNCIAS	39
APÊNDICE I - GABARITO.....	40

Fonte: Elaborado pelo autor.

Outro aspecto essencial a ser considerado é maneira como o material vai ser estruturado. Para Weber e Oliveira (2016) o modo de estruturação do texto é um ponto forte, pois ao dividi-lo em unidades e pequenos parágrafos e/ou subpartes, torna a leitura do material mais leve, facilitando e motivando o leitor para o estudo dos conteúdos. Com esse objetivo de facilitar e motivar os professores para leitura e uso do material, desenvolvemos parágrafos simples e diretos, mas com todas as informações necessárias. Contudo, a estrutura textual, visual e todos os elementos como, quadros e imagens, cores, fonte etc. foram pensados, para tornar o material motivador, de fácil manuseio e compressão das informações aos professores.

6.5 GRUPO NO FACEBOOK

As redes sociais estão cada vez mais presentes no contexto educacional e podem ser utilizadas como meio de divulgação de diversos conteúdos, troca de experiências e compartilhamento de conhecimento, com o uso das suas ferramentas que promovem a comunicação e interação entre seus usuários. Conforme Souza e Schneider (2014), o Facebook, por exemplo, é uma rede social que tem se destacado, “pelas possibilidades pedagógicas aliadas ao uso de suas ferramentas”.

O grande número de acesso por usuários ao Facebook, transforma a rede social em uma “máquina poderosa para se veicular uma notícia, com a vantagem de que é possível se obter um *feedback* (retorno do impacto da informação) quase que imediato” (SILVA; SILVA, 2014, p. 2). Nesse sentido, na busca de divulgar e obter um maior alcance e abrangência da Metodologia 7Cs, criamos um grupo no Facebook³ intitulado “Metodologia 7Cs”.

A escolha por criar um grupo e não uma *fanpage*, se deu em função dos grupos serem voltados para discussões privadas sobre assuntos específicos, neste caso o uso da Metodologia 7Cs. O grupo dispõe de funcionalidades que possibilitam a interação entre os usuários e o compartilhamento de materiais e de informações de forma colaborativa (MAGRIN, 2013). No Quadro 21 apresentamos as funcionalidades do Grupo no Facebook, relacionando aos nossos objetivos com o uso da ferramenta.

³ Grupo da Metodologia 7Cs no Facebook: <https://www.facebook.com/groups/558756151522251/>

Quadro 21- Ferramentas presentes no Grupo e suas funcionalidades

FUNCIONALIDADES	DESCRIÇÃO (MAGRIN,2013)	OBJETIVO DE USO
Mural	Local público de publicações e abertura quando se acessa um determinado grupo.	Postar comunicados importantes e atualizações da Metodologia 7Cs. Os professores podem interagir e responder aos posts diretamente.
Eventos	Permite criar lembretes para os membros do grupo.	Lembrar os professores para divulgar, aplicar e relatar suas experiências com o uso da metodologia.
Documentos	Permite submeter arquivos para <i>download</i> , ou a criação de documentos de texto compartilhados.	Compartilhar materiais atualizados sobre a metodologia, tais como artigos científicos publicados, ou criar um documento de sugestões para o aprimoramento da proposta, de forma compartilhada.
Bate-papo	Ferramenta síncrona similar aos comunicadores instantâneos tradicionais.	Interagir com professores em tempo real, com objetivo de tirar possíveis dúvidas.

Fonte: Adaptado de Magrin (2013).

O grupo foi criado como estratégia para que os professores possam acessar novidades sobre a Metodologia 7Cs, relatar e compartilhar suas experiências ao aplicá-la em sala de aula, apresentando os resultados alcançados. Além disso, abrir a possibilidade de interagir conosco para tirar possíveis dúvidas e propor possíveis sugestões para o aprimoramento da metodologia.

CONSIDERAÇÕES FINAIS

As disciplinas introdutórias à programação exigem que as habilidades e competências sejam trabalhadas de maneira integrada, porém muitos alunos apresentam déficits que acabam por acarretar dificuldades em compreender e construir algoritmos, o que ocasiona em alguns a desmotivação e a desistência, por não conseguirem acompanhar o andamento das disciplinas cada vez mais complexos.

Durante nossa trajetória acadêmica foi possível identificar, por meio das observações realizadas, alguns pontos de maior dificuldade na disciplina de algoritmos pelos alunos, como, por exemplo, a interpretação dos comandos das questões e o encadeamento das informações na construção de algoritmos, que despertou em nós a necessidade de desenvolver uma proposta que pudesse estimular nos alunos o desenvolvimento de habilidades e competências, que agregassem significados a construção do conhecimento, em relação aos conteúdos ministrados.

Dessa maneira concebemos a Metodologia 7Cs que propõe o desenvolvimento de um processo de construção do conhecimento, com uma sequência lógica, para facilitar o entendimento dos conceitos e melhorar o desempenho dos alunos, por meio do estímulo de diferentes habilidades e competências, as quais consideremos importantes para construção de algoritmos. Além disso, percebemos também a importância de criarmos uma metodologia capaz de proporcionar a aprendizagem significativa dos conteúdos. Para tanto, realizamos esse processo com base nas três fases apresentadas na Teoria de Aprendizagem Significativa de Ausubel (2000).

Segundo a teoria estudada a aprendizagem de um novo conhecimento, a partir dos conhecimentos armazenados ou construídos na estrutura cognitiva dos alunos, pode facilitar a compreensão dos conteúdos básicos de algoritmos. Apesar das disciplinas de algoritmos trabalharem, por exemplo, conceitos matemáticos, em que, pressupomos que todos os alunos viram no ensino básico, se torna difícil integrar conhecimentos, os quais eles sentem dificuldades mesmo antes de cursarem a graduação. Entendemos que a construção de algoritmos trata de um conhecimento novo para os alunos, o qual necessita da integração de vários outros conhecimentos. Por isso, a construção da metodologia em fases ocorre para que cada uma agregue informações que construa na estrutura cognitiva dos alunos conhecimentos suficientes, para construção de algoritmos com eficiência.

O teste piloto da Metodologia 7Cs no Programa de Nivelamento em Algoritmos (PNA) no Instituto de Ciências Exatas e Naturais (ICEN) da UFPA, suscitou os primeiros indícios de que ela possui potencial para o ensino e aprendizagem de algoritmos, como: a possibilidade de condução da metodologia por outros professores, ao observarmos o uso dela pelos monitores; o bom desempenho nas atividades realizadas, relacionadas a construção de algoritmos, a partir dos resultados obtidos pelos alunos. Assim, apesar de não podermos afirmar categoricamente o quanto a metodologia influenciou na aprendizagem, podemos ressaltar a aceitação e facilidade de compreensão desta pelos alunos.

Ao aplicarmos a Metodologia 7Cs na disciplina de Algoritmos CBCC da UFPA, verificamos que a metodologia é viável para ser utilizada em contexto real de sala de aula e o desempenho e a boa aceitação por parte dos estudantes apontam para o fato de que a Metodologia 7Cs pode contribuir no processo de ensino e aprendizagem, em disciplinas introdutórias de programação. Alguns indícios para tais afirmações foram: o uso da metodologia, sem maiores dificuldades pela professora da disciplina Algoritmos, o que nos sugere que provavelmente outros professores conseguirão utilizá-la em suas disciplinas; o bom desempenho na construção de algoritmos, por parte dos estudantes, com o qual algumas vezes fomos surpreendidos positivamente, pela agilidade de alguns alunos ao desenvolverem algoritmos que consideramos complexos; o estímulo a atividades colaborativas, que consideramos ter potencializado a aprendizagem dos conteúdos, à medida em que os alunos compartilharam e agregaram conhecimentos em sua estrutura cognitiva; e o superior rendimento dos alunos na disciplina, se compararmos com as turmas anteriores, nas quais realizamos a observação participante. Essa turma, em sua maioria, estava com nível de compreensão elevado ao longo da disciplina, fato refletido nas boas notas e avanço dos conteúdos de forma rápida.

Diante das lacunas percebidas na literatura em termos de metodologias voltadas ao ensino e aprendizagem de algoritmos, destacamos que a importância da Metodologia 7Cs está em: buscar desenvolver habilidades e competências primárias a construção de algoritmos por meio da prática como, por exemplo, o estímulo à análise e interpretação de comandos; promover atividades colaborativas que potencializam o processo de aprendizagem dos alunos e; possibilitar ao alunos desenvolverem o processo criativo, por meio da autonomia em criar, com base em seus conhecimentos, comandos de questões. Além disso, por meio das aplicações, percebemos que a metodologia pode tornar as aulas

mais dinâmicas e motivadoras, por permitir elaborar diferentes possibilidades e formas de uso, sendo flexível e adaptável a outras propostas.

A Metodologia 7Cs, assim como muitas outras criadas, não alcança ou soluciona todos os percalços vivenciados nas disciplinas voltadas à programação. Existem dificuldades e lacunas que não são tratadas diretamente pela metodologia, mas cujo tratamento auxiliaria ainda mais o bom desempenho nas disciplinas, como por exemplo, dificuldades relacionadas a conceitos matemáticos básicos. Ainda assim, por tratar de forma diferenciada elementos essenciais à construção de algoritmos, trata-se de uma proposta que tem um grande potencial para reduzir as dificuldades na compreensão dos conceitos básicos e no desenvolvimento de habilidades associadas em disciplinas de algoritmos, se utilizada de forma correta, seguindo a lógica de construção de conhecimentos e promovendo aprendizagem significativa.

De acordo com a proposta deste trabalho de pesquisa, consideramos ter conseguido alcançar os objetivos específicos propostos inicialmente ao: identificar por meio da literatura e observação participante as dificuldades de ensino e de aprendizagem em disciplinas voltadas a programação; conceber e desenvolver uma metodologia a partir das reais necessidades e contexto vivenciado por muitos alunos em cursos da área da computação; realizar um teste piloto como o foco em verificar a viabilidade de aplicação e recolher elementos para o aprimoramento da metodologia; e, por fim, ao aplicar a metodologia em um contexto real de sala de aula, na disciplina Algoritmos no CBCC da UFPA.

Como ressaltamos anteriormente, sabemos que não existe uma metodologia que possa de fato solucionar e alcançar as diversas dificuldades dos alunos, mas entendemos a importância em buscar estratégias e metodologias que possam auxiliar no processo de ensino e aprendizagem. O objetivo geral do trabalho desenvolvido foi conceber uma metodologia de ensino e de aprendizagem para professores e alunos, com o intuito de minimizar as dificuldades na compreensão dos conteúdos de disciplinas introdutórias à programação, para que assim pudéssemos contribuir na diminuição de recorrentes e elevados índices de reprovações e evasão. Acreditamos ter tido sucesso nessa empreitada, com a concepção da Metodologia 7Cs.

Para que a Metodologia 7Cs pudesse ser acessada e utilizada com mais abrangência, elaboramos um material de orientação, que permitirá uma compreensão mais rápida, por parte dos professores, dos princípios e funcionamento da metodologia.

Evidentemente, apresentamos apenas um exemplo ilustrativo de aplicação, em que são criadas apenas duas listas de exercícios, que tratam somente de duas estruturas de algoritmos. A proposta consiste em que os professores, tendo entendido a concepção central da metodologia, possam se sentir motivados para construir suas próprias atividades e indicações acerca de quaisquer das estruturas trabalhadas nos cursos de algoritmos.

Finalmente, durante a concepção e desenvolvimento da Metodologia 7Cs, percebemos várias possibilidades e potencialidades para sua aplicação em outros contextos e diferentes conteúdos. Pretendemos também evoluir a metodologia, em busca de contribuir e facilitar a compreensão dos conteúdos de outras disciplinas e de diferentes cursos de graduação. Além disso, temos claro que a Metodologia 7Cs apresenta alguns elementos característicos de uma metodologia ativa (atividades colaborativas, elementos de gamificação, incentivo a que os estudantes proponham próprios problemas, possibilidade de utilização da metodologia de forma autônoma pelos alunos) e que, apesar de não ter sido objetivo deste trabalho, pretendemos ainda, em um futuro breve, propor incrementos à estrutura da metodologia, de modo a que os alunos passem a ter um protagonismo ainda maior em seus processos de aprendizagem e que possamos caracterizar a Metodologia 7Cs, de forma mais específica, como uma metodologia ativa.

REFERÊNCIAS BIBLIOGRÁFICAS

ABIB, G.; HOPPEN, N.; HAYASHI JÚNIOR, P. H. Observação Participante em Estudos de Administração da Informação no Brasil. **RAE-Revista de Administração de Empresas**. São Paulo, v. 53, n. 6, p. 604-616, 2013.

ALENCAR, E.; FLEITH, Denise. **Criatividade: múltiplas perspectivas**. 3 ed. Brasília: Editora da Universidade de Brasília, 2003.

AMBRÓSIO, A. P. L. et al. Programação de Computadores: compreender as dificuldades de aprendizagem dos alunos. **Revista Galego-Portuguesa de Psicoloxía e Educación**, v. 19, n. 1, p. 185-197, 2011.

AUSUBEL, David P. **Aquisição e retenção de conhecimentos**. Lisboa: Plátano Edições Técnicas. Tradução do original *The acquisition and retention of knowledge*, 2000.

BARCELOS et al. O uso de mobile learning no ensino de algoritmos. **RENOTE - Revista Novas Tecnologias na Educação**, v. 7, n. 3, 2009.

BRANCO, N. W. C.; SCHUVARTZ, A. A. Ferramenta Computacional de Apoio ao Processo de Ensino-Aprendizagem dos Fundamentos de Programação de Computadores. In: Simpósio Brasileiro de Informática na Educação (SBIE), v. 28, 2007, São Paulo, SP. **Anais...** 2007, p. 520-528.

BESEMER, S.; TREFFINGER, D. **Analysys of Creative Products**: Review and Synthesis. *The Journal of Creative Behavior*, v. 15, n. 3, p. 158-178, 1981.

BERSSANETTE, J. H. **Ensino de programação de computadores: uma proposta de abordagem prática baseada em Ausubel**. Ponta Grossa, PR: UTEFP, 2016. Originalmente apresentada como dissertação de mestrado, Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2016.

CAMPOS, R. L. B. L. ERM2C methodology: The new strategy to improve the learning/teaching process in algorithm. In: 8º CONGRESSO INTERNACIONAL DE GESTÃO DE TECNOLOGIA E SISTEMAS DE INFORMAÇÃO – CONTECSI, v.8, 2011, São Paulo, SP. **Anais...** 2011, p. 1-14.

CAMPOS, R. L. B. L. ERM2C: Uma metodologia para melhoria do ensino-aprendizado de lógica de programação. In: VIII WORKSHOP DE EDUCAÇÃO E INFORMÁTICA, v.3, 2010, Maceió, AL. **Anais...** 2010, p. 1-3.

CAMPOS, R. L. B. L. ERM2C: Uma proposta de metodologia para melhoria do ensino-aprendizado de lógica de programação. In: XI CONGRESSO CHILENO DE EDUCAÇÃO SUPERIOREM COMPUTAÇÃO – CCESC, v. 11, 2009, Santiago, Chile. **Anais...** 2009, p. 1-8.

DELGADO, C. et al. Uma Abordagem Pedagógica para a Iniciação ao Estudo de Algoritmos. In: XII WORKSHOP DE EDUCAÇÃO EM COMPUTAÇÃO, v.5, 2004, Salvador, BA. **Anais...** 2004, p. 72-87.

GIL, A. C. **Métodos e Técnicas de Pesquisa Social**. 6ª Edição. São Paulo: Atlas. 2008. Disponível em: <https://ayanrafael.files.wordpress.com/2011/08/gil-a-c-mc3a9todos-e-tc3a9cnicas-de-pesquisa-social.pdf>. Acesso em: 10 dez. 2018.

GIRAFFA, L. M. M.; MORA, M. C. Evasão na Disciplina de Algoritmo e Programação: Um Estudo a Partir dos Fatores Intervenientes na Perspectiva do Aluno. In: III CONFERENCIA SOBRE EL ABANDONO EN LA EDUCACIÓN SUPERIOR (III CLABES), v. 3, 2013, Espanha. **Anais...** 2013.

HORN; SALVENDY. **Criatividade produto: modelo conceitual, medição e características**. In: Questões teóricas em Ciência da Ergonomia. v. 7, n. 4, 2006.

LIMA, Á. C.; SOUSA, D. F. Experiência no Programa Institucional de Bolsas de Iniciação à Docência (PIBID): Desenvolvimento do Raciocínio Lógico e Algoritmo na Educação Básica. In: I WORKSHOP DE ENSINO EM PENSAMENTO COMPUTACIONAL ALGORITMOS E PROGRAMAÇÃO INFORMÁTICA NA ESCOLA (WAIGPROG), v1, 2015, Maceió, AL. **Anais...** 2015, p. 1290-1299.

KNELLER, G. **Arte e ciência da criatividade**. 5. ed. São Paulo: Ibrasa, 1978.

LAVILLE, C.; JEAN, D. **A Construção do Saber: Manual de metodologia da pesquisa em ciências humanas**. Porto Alegre: Artmed; Belo Horizonte: Editora UFMG, 1999.

MOURÃO, A. Uma proposta da eficiência do uso da metodologia ativa baseada em problemas, utilizando dojo de programação, aplicada na disciplina de lógica de programação. In: WORKSHOP DE INFORMÁTICA NA ESCOLA, v. 23, 2017, Recife, PE. **Anais...** 2017 p. 667-676.

MALCHER, M. A.; ELIASQUEVICI, M.; VELOSO; J. M. **Metodologia de Validação de Processos e Produtos Educacionais**. 3 out. 2018. 44 slides. Material apresentado para a disciplina de Metodologia de Validação de Processos e Produtos Educacionais no Programa de Pós-Graduação Criatividade e Inovação de Metodologias de Ensino Superior da Universidade Federal do Pará.

MASETTO, M. **Inovação no ensino superior**. São Paulo: Edições Loyola, 2012.

MASETTO, M. Técnicas para o desenvolvimento da aprendizagem em aula. In: MASETTO, M. **Competência pedagógica do professor universitário**. 3. ed. São Paulo: Summus, 2015, p. 97-157.

MOREIRA, M. A. **Aprendizagem Significativa: a teoria e textos complementares**. São Paulo: Editora Livraria da Física, 2011.

MOREIRA, M. A. **Aprendizagem Significativa em Mapas Conceituais: Meaningful learning in concept maps**. Disponível em: <https://www.if.ufrgs.br/~moreira/apsigmapasport.pdf>. Acesso em: 10 jan. 2019.

MOREIRA, M. A; MASINI, E. F. S. **Aprendizagem Significativa: A teoria de David Ausubel.** São Paulo: Moraes, 1982.

MOREIRA, A. M.; NARDI R. O mestrado profissional na área de Ensino de Ciências e Matemática: alguns esclarecimentos. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 2, n. 3, set./dez. 2009.

MENEZES, E. T. de; SANTOS, T. H. dos. Verbete transversalidade. **Dicionário Interativo da Educação Brasileira - Educabrazil.** São Paulo: Midiamix, 2001. Disponível em: <http://www.educabrazil.com.br/transversalidade/>. Acesso em: 16 de abr. 2018.

MORAIS, M. de F.. A avaliação da criatividade: a opção pelos produtos criativos. **Revista RecreArte**, Santiago de Compostela – Espanha, n. 4, p. 1-13, 2005.

NOBRE, I. A. M.; MENEZES, C. S. Suporte à Cooperação em um Ambiente de Aprendizagem para Programação (SAmbA). In: XII SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, v. 12, 2001, Vitória, ES. **Anais...** 2001, p. 337-347.

PETRY, P. G. **Um Sistema Para o Ensino e Aprendizagem de Algoritmos Utilizando um Companheiro de Aprendizagem Colaborativo.** Florianópolis, SC: UFSC, 2015. Originalmente apresentada como dissertação de mestrado, Universidade Federal de Santa Catarina, 2005.

PINTO, A. S.; BUENO, MARCILENE R. P.; SILVA, M. A. F. A.; SELLMANN, M. Z.; KOEHLER, S. M. F. **Inovação Didática** - Projeto de Reflexão e Aplicação de Metodologias Ativas de Aprendizagem no Ensino Superior: Uma Experiência com “Peer Instruction”. Janus, América do Norte, 2012.

PELIZZARI, A. et al. Teoria da Aprendizagem Significativa Segundo Ausubel. **Revista PEC**, v.2, n1, p. 37-42, 2002.

PRETI, O. **Produção de Material Didático Impresso: Orientações Técnicas e Pedagógicas.** Cuiabá. Editora UFMT, 2010.

PROENÇA, W. L. O Método da Observação Participante: Contribuições e aplicabilidade para pesquisas no campo religioso brasileiro. **Revista Aulas**, v. 10, n.4, p.1-24, 2007.

RAPKIEWICZ, E. C. et. al. Estratégias Pedagógicas no Ensino de Algoritmos e Programação Associadas ao Uso de Jogos Educacionais. **Revista Renote** – Novas Tecnologias na Educação, v.4, n.2, p. 1-11, 2006. Disponível em: <http://seer.ufrgs.br/renote/article/view/14284>. Acesso em: 14 set. 2018.

ROCHA, S.P; FERREIRA, B; MONTEIRO, D; NUNES, C. S.D;GÓES, N. C. H. Ensino e Aprendizagem de Programação: Análise da aplicação de Proposta Metodológica Baseada no Sistema Personalizado de Ensino. **Revista Renote** – Novas Tecnologias na Educação, v.8, n3, p. 1-11, 2010. Disponível em: <http://seer.ufrgs.br/renote/article/view/18061/10649>. Acesso em: 10 jan. 2019.

SAMPIERI, R.; COLLADO, C.; LUCIO, M. **Metodologia de Pesquisa**. 5. ed. Porto Alegre: Penso, 2013.

SILVA, V., & SILVA, A. O papel do Facebook na divulgação científica de notícias relacionadas ao meio ambiente. **Revista Extraprensa**, 8(1), 177-192, 2014.

SOUSA, S. de O. **Aprendizagem Baseada em Problemas (PBL – Problem-based Learning)**: Estratégia para o Ensino e Aprendizagem de Algoritmos e Conteúdos Computacionais. Presidente Prudente, SP: UEP, 2011. Originalmente apresentada como dissertação de mestrado, Universidade Estadual Paulista, 2011.

SOUZA, A. A. N.; SCHNEIDER, H. N. O Facebook como espaço de interação, colaboração e aprendizagem: uma reflexão sob a perspectiva discente. 3º Congresso Brasileiro de Informática na Educação - 20ª Workshop de Informática na Escola. **Anais ...**p. 342-351, 2014.

SCHWENGBER, E. et al. Avaliação subjetiva da criatividade em produtos da atividade projetual. **Revista Educação**, v. 10, n. 1, p. 61-77, 2015.

TIMMERMANN, G. L. K. **O Desafio de Aprender e Ensinar Algoritmos: Mediações que Professores e Alunos Estabelecem com o Conteúdo no Ensino Superior**. Ijuí, RS: UNIJUÍ, 2015. Originalmente apresentada como dissertação de mestrado, Universidade Regional do Noroeste do Estado do Rio Grande do Sul, 2015.

TORRE, S. de la. **Dialogando com a criatividade**: da identificação à criatividade paradoxal (tradução Cristina Mendes Rodríguez). São Paulo: Madras, 2005.

VIEIRA, E. C.; LIMA JUNIOR, A. T. de; VIEIRA, P. de P. V. **Dificuldades no Processo de Aprendizagem de Algoritmos**: uma Análise dos Resultados na Disciplina de AL1 do Curso de Sistemas de Informação da FAETERJ. Campus Paracambi, v. 10, n.1, p.1-15, 2015.

WEBER, D. J.; OLIVEIRA, L. R. Materiais Didáticos para Educação a Distância: Observando Layouts. **Revista Científica em Educação a Distância**, v. 6, n. 1, p. 152-167, 2016.

WECHSLER, S. M. **Criatividade**: Descobrendo e Encorajando: contribuições teóricas e práticas para as mais diversas áreas. São Paulo: Edições Loyola, 1993.

APÊNDICE I – ROTEIRO DE OBSERVAÇÃO

ROTEIRO DE OBSERVAÇÃO

Professor (a):

Disciplina: Algoritmo e Programação

Conteúdo da aula: _____

Turno: Tarde

Horário da Aula: Início:__:__ e Fim:__:__

Data da observação: __/__/__

1. Existe coerência entre o proposto e o que foi realizado na prática em sala de aula?

Sim () Não ()

Obs.: _____

2. Quais recursos didáticos utilizados?

() Quadro

() Slides

() Computador

() Datashow

() Jogos

Outros: _____

Obs.: _____

3. Os recursos utilizados são adequados ao conteúdo proposto?

Sim () Não ()

Obs.: _____

4. As atividades propostas para a turma foi de natureza individual ou coletiva?

Individual ()

Coletiva

Descrição: _____

5. As atividades e os problemas propostos são desafiadores e proveitosos para a aprendizagem dos discentes?

Sim () Não ()

Obs.: _____

6. Para todos os alunos as atividades surtiram o mesmo efeito? Ou alguns compreenderam de maneira mais fácil e, outros, mais difícil?

Sim () Não ()

Obs.: _____

7. Quais as principais dificuldades dos alunos durante a aula e atividades propostas?

() Compreensão do conteúdo

() Interpretação do comando

() Conceitos matemáticos

Organização das ideias

Outra:_____.

Obs.:_____.

8. Os alunos ao solicitarem ajuda são rapidamente atendidos pela professora e monitores?

Sim Não

Obs.:_____.

9. Existe uma boa relação entre os alunos, professora e monitores?

Sim Não

Obs.:_____.

10. O período de duração para realizar as atividades foi suficiente para os alunos fazerem?

Anotações

Exporem as dúvidas

Debaterem

Resolverem todos os problemas

Resolverem quase todos os problemas

Obs.:_____.

11. Durante o processo utilizou-se de alguma atividade lúdica? Se sim, de que maneira foi recebida pelos alunos?

Sim Não

Obs.:_____.

12. Os conteúdos são contextualizados com a realidade sociocultural dos alunos?

Sim Não

Obs.:_____.

13. Os alunos cooperam uns com os outros?

Sim Não

Obs.:_____.

APÊNDICE II - LISTAS DE EXERCÍCIOS - ESTRUTURA REPETIÇÃO (PRIMEIRO TESTE)

LISTA DE EXERCÍCIOS 10 (REPETIÇÃO (7Cs))



1) A partir de sua **compreensão**, escreva o que o programa faz?

```

Program Compreender;
var
  i, n, termo, termo_1, termo_2: integer;
begin
  termo_2 := 0;
  termo_1 := 1;
  repeat
    write('Quantos termos? ');
    readln (n);
  until n > 0;
  for i := 1 to n do
    begin case i of
      1: write(termo_2:4);
      2: write(termo_1:4);
      else
        termo := termo_1 + termo_2;
        write(termo:4);
        termo_2 := termo_1;
        termo_1 := termo;
      end;
    end;
  end.

```

2) Um funcionário de uma empresa recebe aumento salarial anualmente. Sabe-se que:

- Esse funcionário foi contratado em 2000, com salário inicial de R\$ 500.00;
- Em 2005 recebeu aumento de 0,5% sobre seu salário inicial;
- A partir de 2012 (inclusive), os aumentos salariais sempre correspondem ao dobro do percentual do ano anterior.

Faça um programa que determine o salário atual desse funcionário.

3) Escreva um algoritmo que realize a potência de A (número inteiro) por B (Número inteiro e positivo), ou seja A elevado a B, por meio de multiplicações sucessivas. Esses dois valores são passados pelo usuário por meio do teclado. Com base no comando, **complete** o algoritmo a seguir:

```

Programa Completar;
Var
  a, b, i: integer;
Begin
  Writeln (' Digite dois números ');
  readln (a);
  then writeln ('não há como fazer potência')
  if b =
    then pot :=1
    else
      pot := pot* ;
if b >= 0
  then writeln (a, ' elevado a', b, '=', pot);
End.

```

4) Elabore um algoritmo que entre com números inteiros enquanto forem positivos e imprima a média dos números digitados. De acordo com o comando, **compatibilize** (reorganizar) o algoritmo a seguir:

```

Program Compatibilizar
  Cont: interger;
Var
  Total, media, num: real;
Begin
  Write ('digite um número');
  Cont:= 1;
  Total:= num;
  While (num>0) do
    Read (num);
    Begin
      Writeln ('digite um número:');
      If (num>0) then
        Total:= total + num;
        Begin
          Cont := Cont + 1;
          Read (num);
        End;
    End;
  media := total/cont;
  writeln (' A media dos valores digitados é:', media: 4:2);
End.

```

5) Faça um algoritmo que leia a idade de um número indeterminado de pessoas. Quando o usuário digitar uma idade ≤ 0 , o programa deve parar com a entrada de dados. Em seguida o algoritmo deverá apresentar:

- A idade do mais velho
- A idade do mais novo
- A quantidade de adolescentes (de 12 a 17 anos)

Partindo do comando, encontre os possíveis erros, no algoritmo a seguir, e os corrija:

```

Program Corrigir;
Var
  Idade, menor, maior, cont: integer;
Begin
  Repeat
  Writeln ('Digite a idade');
  Read ('idade');
  if (idade <> 0) then
    begin
      if (idade < maior) then
        maior := idade;
      if (idade > maior) then
        maior = idade
      if (idade >= 12) or (idade <= 17) then
        cont := cont + 1;
    end;
  until (idade <= 0);
  writeln ('A idade do mais velho é de', menor, 'anos');
  Writeln ('A idade do mais novo é de', maior, 'anos');
  Writeln (' Numero de adolescentes (13 a 17 anos):', idade);
End.

```

6) **Construa** um algoritmo que leia o número de gols marcados pelo Brasil e o número de gols marcados pela Argentina, escrevendo o nome do time vitorioso ou a palavra EMPATE. A seguir, escreva a mensagem “Novo Jogo? 1. Sim 2. Não” e solicitar uma resposta. Se a resposta for 1 o algoritmo deve ser executado novamente solicitando o número de gols marcados pelos times em uma nova partida, caso contrário deve ser encerrado escrevendo:

- Quantos Jogos fizeram parte da estatística
- O número de vitórias do Brasil
- O número de vitórias da Argentina
- O número de empates

7) Foi realizada uma pesquisa entre os habitantes de uma região e coletados os dados de altura e sexo (0=masc, 1= fem) das pessoas. Faça um algoritmo que leia 50 dados diferentes e informe:

- A maior e a menor altura encontradas
- A média de altura das mulheres
- A média de altura da população
- O percentual de homens na população

Para realização da questão serão formados grupos que **trabalharam de forma colaborativa** para solucionar o problema.

LISTA DE EXERCÍCIOS 11 (REPETIÇÃO (7Cs))



1) Analisem e interpretem os trechos de códigos a seguir, identificando em sua resposta o que **compreendem** e se são ou não equivalentes.

```
a) i := 1; a:= 2; b:= 1;
While i < 4 do
Begin
i := i + 1;
a := a * b;
Writeln (a, ' ', i);
End;
```

```
a1) i := 1; a := 2; b :=1;
Repeat
i := i + 1;
a := a * b;
writeln (a, ' ', i);
Until i > 3;
```

```
b) i := 1; j:= 20;
While i < 4 do
Begin
j := j - i;
i := i +1;
End; Writeln (j);
```

```
b1) j := 20;
For i := 1 to 4 do
j := j - i;
Writeln (j);
```

2) Uma das maneiras de se conseguir a raiz quadrada de um número é subtrair do número os ímpares consecutivos a partir de 1, até que o resultado da subtração seja menor ou igual a zero. O número de vezes que se conseguir fazer a subtração corresponde a raiz quadrada exata (resultado 0) ou aproximada do número (resultado negativo). **Conceba** um programa que dado um número inteiro positivo, calculará e exibirá a sua raiz.

Ex: Raiz de 16

$16 - 1 = 15 - 3 = 12 - 5 = 7 - 7 = 0 \rightarrow$ A raiz quadrada de $16 = 4$

3) Faça um algoritmo que calcule e escreva o valor da seguinte somatória:

$S = 1 + 2/9 + 4/27 + 8/81 + 16/243$

Com base no comando, **complete** o algoritmo a seguir:

```
Program Completar; Var
i, j: integer soma: real
Begin
j := 2;
k := 9;
writeln (soma); for i := 1 to 4
begin
soma := soma + j
writeln (j, '/', k, '=', soma); j := j * 2;
end;
writeln ( 'Total:');
End.
```

4) Preparar um algoritmo para calcular 100 termos da seguinte série:

$1 - 1/2 + 1/4 - 1/6 + 1/8 - 1/10 + 1/12 - \dots$

De acordo com o comando, **compatibilize** (reorganizar) o algoritmo a seguir:

```

Program Compatibilizar;
Var
i, a: integer; s: real;
s := 1;
Begin
  if i mod 2 <> 0 for i := 1 to 99 do
    begin
      then s := s + (1/(i * 2));
        end;
      else
        s := s - (1/(i * 2))
        writeln (s);
      End.

```

5) Sendo $H = 1 + 1/2 + 1/3 + 1/4 + \dots + 1/N$, faça um algoritmo para calcular H. Partindo do comando, encontre os possíveis erros, no algoritmo a seguir, e os **corrija**:

```

Program Corrigir;
Var
i, n: real;
h : integer; Begin
Writeln ('digite um numero'); Roud (n);
H = 1;
For n := 2 to h do N := h + 1/i;
  Writeln (H); End.

```

6) **Construa** um algoritmo que leia a idade, a altura e o peso de várias pessoas, calcule e imprima:

- A quantidade de pessoas com idade superior a 60 anos.
- A média das alturas das pessoas com idade entre 10 e 30 anos.
- A porcentagem de pessoas com peso inferior a 40 quilos entre todas as pessoas analisadas.

Atenção: quando a idade for igual a zero termina a leitura dos dados.

7) Desenhar um triângulo feito de m linhas de asteriscos (*) alinhadas pela esquerda, em que a primeira linha contém um asterisco, a segunda dois, a terceira três e assim sucessivamente até que o número de asteriscos seja igual à m.

Exemplo: para $m = 6$ a saída do programa deve ser:

```

*
**
***
****
*****
*****

```

Para realização da questão serão formados grupos que **trabalharam de forma colaborativa** para solucionar o problema.

APÊNDICE III – LISTA DE EXERCÍCIO NA ESTRUTURA VETOR (NOVO TESTE)

LISTA DE EXERCÍCIOS 12 (Vetores (7Cs)) (Fase Descoberta)



1) A partir de sua **compreensão**, escreva o que os programas a e b fazem?

a) Program **Compreender**; var
v: array [1..20] of string; nome : string;
i, cont : integer;
Begin
cont := 0;
for i := 1 to 20 do begin
write (' Informe o nome', i, ':'); readln (nome);
if upcase (nome [1]) = 'S' then begin
cont := cont + 1; v[cont] := nome;
end; end;
writeln ('xxx', cont, 'nomes'); End.

b) Program **Compreender**; var
v: array [1..20] of integer; i, aux: integer;
Begin
for i := 1 to 20 do begin
writeln ('Informe o elemento', i, ':'); readln (v [i]);
end;
for i := 1 to 10 do begin
aux := v [i];
v [i] := v [21 - i];
v [21 - i] := aux; end;
for i := 1 to 20 do write (v [i] : 4);
End.

2) Elaborar um algoritmo que leia um vetor de 30 números reais e informe: quantos números lidos são iguais a 30; quantos são maiores que a média; e quantos são iguais a média.

3) Escrever um algoritmo/programa em Pascal que lê um vetor de seis elementos e o escreva. Conte, a seguir, quantos valores de V são negativos e escreva esta informação.

4) Faça um programa em Pascal que leia um vetor de 10 elementos numéricos e escreva apenas as posições pares do vetor.

5) Faça um programa em pascal que leia dois vetores de 5 posições, calcule um vetor com a soma dos elementos em índices iguais e apresente ao usuário.

6) Elaborar um algoritmo que leia um conjunto de 10 elementos do tipo inteiro, calcule e imprima o valor de S, onde:

$$S = (A1 - A10)^2 + (A2 - A9)^2 + (A3 - A8)^2 + (A4 - A7)^2 + (A5 - A11)^2.$$

APÊNDICE IV - LISTAS DE EXERCÍCIOS - ESTRUTURA MATRIZES (TESTE FINAL)

LISTA DE EXERCÍCIOS 14 (Matrizes (7Cs)) (Fase Descoberta)



1) A partir de sua **compreensão**, escreva o que os programas a e b fazem?

a)

```
Program Comprender1;
Var
matriz: array[1..2, 1..2] of integer; i, j: integer;
Begin
  for i:=1 to 2 do
    For j:=1 to 2 do begin
      writeln('Digite um numero para a posição ',i,', ',j); read(matriz[i][j]);
    End; clrscr;
  for i:=1 to 2 do begin
    for j:=1 to 2 do begin
      if i+j = 3 then write(' ',matriz[i][j], ' ') else write(' - ');
    end; writeln;
  end;
End.
```

b)

```
Program Comprender 1; Var
matriz: array[1..7, 1..8] of integer; l, c: integer;
Begin
  for l:=1 to 7 do begin
    for c:=1 to 8 do begin
      matriz[l][c] := l + c;
      write(matriz[l][c]:4); end;
    writeln; end;
End.
```

2) Desenvolva um programa em Pascal que implemente uma matriz 3x3 de números reais, leia os valores e depois apresente para o usuário apenas os elementos da diagonal principal.

3) Desenvolva um programa em Pascal para armazenar a agenda do usuário. Portanto, o programa deve receber os seguintes dados pessoais referente aos contatos do usuário: nome, endereço, CEP, Bairro e Telefone. O tipo de dado que essa matriz deve receber é do tipo String. Considere que o usuário poderá ter apenas 7 contatos na agenda.

4) Desenvolva um programa em que leia uma matriz de ordem 4. Após coloque em um vetor apenas os elementos da diagonal principal. Escreva a matriz e o vetor.

5) Criar um algoritmo que possa armazenar as alturas de dez atletas de cinco delegações que participarão dos jogos de verão. Imprimir a maior altura de cada delegação.

LISTA DE EXERCÍCIOS 15 (Matrizes (7Cs))
(Fase Aperfeiçoamento)



1) Faça um programa que gere a seguinte matriz:

```
1 1 1 1 1 1
1 2 2 2 2 1
1 2 3 3 2 1
1 2 3 3 2 1
1 2 2 2 2 1
1 1 1 1 1 1
```

Com base no comando, complete o algoritmo a seguir:

```
Program Completar;
Var
mat: array [1.. ] of integer; c: integer;
Begin
  For l := 1 to 6 do Begin
    if (l = 1) or (l = ) then mat [l,c] := 1;
    if ((l >= 2) and (l <= 5)) and ((c >=2) (c <= 5));
    if ((l >=) and (l <= 4)) and ((c >= 3) and ( c <=4)) then
      mat [l] :=3; end;
    for l := 1 to 6 Begin
      Writeln;
      For c:= 1 to 6 do Write (mat [l,c], “);
    end;
  End.
```

2) Desenvolva um algoritmo que leia do usuário um conjunto de números de uma matriz quadrada de ordem 3 e apresente ao usuário os números armazenados. Regra de constituição de matriz quadrada: número de colunas = número de linhas. De acordo com o comando, compatibilize (reorganizar) o algoritmo a seguir:

```
Program Compatibilizar;
  matQ : array [1..3,1..3] of integer; Begin
  Var
  for i:=1 to 3 do Begin
    i, j: integer;
    Writeln('Apresentação da Matriz 3x3'); Begin
      Write('Informe o valor da linha ',i,' coluna ', j,' :');
      Write(['',matQ[i,j],']');
      read(matQ[i,j]); end;
  End.
  for i:=1 to 3 do for j:=1 to 3 do
    Begin end;
  for j:=1 to 3 do end;
```

3) Desenvolva um algoritmo em pascal que monte uma nota fiscal, para isso utilize a estrutura de matriz, na primeira coluna deve armazenar a quantidade de produto solicitado, na segunda linha o valor unitário e na terceira linha o valor total por produto. Considere que a nota fiscal pode ter 10 linhas e o usuário informará na primeira interação os 10 produtos.

Partindo do comando, encontre os possíveis erros, no algoritmo a seguir, e os **Corrija**:

```

Program Corrigir ;
Var
nota: array [1..15, 1..2] of inteiro; i, j: real;
Begin
  Writeln('Nota fiscal'); for i:=1 to 13 do
    for j:=1 to 3 do Case j of
      1: Begin
          Writeln('=====');
          Write('Informe a quantidade : '); readln(nota[i,i]);
        end; 2: Begin
          Write('Informe o valor unitário : '); readln(nota[i,j]);
        end; 4: Begin
          nota[i,j] = 1; nota[i,j] = nota[i,j+1]+nota[i,j +2]; end;
    clrscr;
    writeln('||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||');
    writeln('Nota fiscal final');
    writeln (' Quantidade || Valor Unitário || Valor total '); for i:=1 to
    15 do
      Begin
        for j:=1 to 2 do Begin
          Case i of
            1: Write(' ',nota[i,j]:2:0,' ||');
            3: Write(' ',nota[i,j]:2:2,' ||');
            3: Write(' ',nota[i,j]:2:2); end;
          end;
        end;
      End.

```

LISTA DE EXERCÍCIOS (Matrizes (7Cs))
(Fase Consolidação)



1) **Construir** um programa em Pascal que entre com valores para duas matrizes inteiras de ordem 5, gere e imprima a matriz diferença.

2) O Triângulo de Pascal é formado por uma composição especial de números inteiros. Cada um dos números do triângulo é formado pela soma de dois números da linha anterior: um imediatamente acima e o outro, acima e a esquerda. **Construir** um programa em Pascal para ler um número inteiro positivo que representa a altura de um Triângulo de Pascal, gerar e imprimir os valores desse Triângulo no monitor de vídeo do computador (a altura máxima da matriz deve ser 50). Veja um exemplo de execução do programa:

Altura = 7

```
Linha 0 1
Linha 1 1 1
Linha 2 1 2 1
Linha 3 1 3 3 1
Linha 4 1 4 6 4 1
Linha 5 1 5 10 10 5 1
Linha 6 1 6 15 20 15 6 1
Linha 7 1 7 21 35 35 21 7 1
```

ANEXO I – QUESTÕES DA FASE APERFEIÇOAMENTO NO KAHOOT!

QUESTÕES DA DIMENSÃO COMPLETAR

Qual função deve ser concluída para que sejam retornados os números de 0 a 2?

25

```
for x in ... (0, 3):
    print(x)
```

00 Respostas

▲ espaço ◆ volta

● alcance ■ escala

kahoot.it PIN do jogo: **1278848**

Conclua ou programa para que a mesquita seja um número maior, igual ou menor que zero.

56.

```
if soma > 0:
    print("Maior que zero")
# ...
print("Igual a zero")
else:
    print("Menor que zero")
```

00 Respostas

▲ elif soma = 0 ◆ print soma = 0

● else soma == 0: ■ elif soma == 0:

kahoot.it PIN do jogo: **1278848**

Para completar o programa que demonstra números negativos e positivos, o que está faltando?

55

```
num = int
if num = 0:
    if num > 0
        print("Positivo.")
    else:
        print("Negativo.")
else:
    print("Nulo.")
```

00 Respostas

▲ A linha 1 deve vir acompanhada de (input []).

◆ Uma alternativa 1 e hum ! na linha 2 para um número diferente de 0.

● Uma alternativa 1 e 2 e um : depois do 0 na linha 3.

■ Não há nada para completar.

kahoot.it PIN do jogo: **1278848**

Pedro Quer Saber ao Digitar Dois Números **um e b** Qual o Maior Deles. Complete o algoritmo.



25

```
a, b = input("Digite os valores a e b separados por espaço: ").split(" ")
a = int(a)
b = int(b)

if a > b:
    print("O maior número é {}".format(a))
elif a < b:
    print("O maior número é {}".format(b))
else:
    print("Os dois números são iguais.")
```

Pular

00

Respostas

▲ elif a! = b:

◆ caso contrário b> a:

● elif a == b:

■ System.out.print ("Olá, mundo")

kahoot.it PIN do jogo: 1278848

O que está faltando no programa para que ele não gere um loop infinito e funcione corretamente?



23

```
i = 0
while i < 10:
    print(i)
    #...
```

Pular

00

Respostas

▲ i ++

◆ i += 1

● i -= 1

■ pausa

kahoot.it PIN do jogo: 1278848

Complete ou o algoritmo para o resultado imprimido seja [0, 1, 2, 3, 4, 5, 6, 7, 8, 9].



23

```
L = []
for i in range(10):
    #...
print(L)
```

Pular

00

Respostas

▲ L = L + i

◆ L = [i]

● L.inserção (i)

■ L.append (i)

kahoot.it PIN do jogo: 1278848

Para Que a Variável **um** SEJA trocada de valor com uma Variável **b** , O Que Precisa Ser completado?

55

```
a = 2
b = 4

aux = a
#...
b = aux
```

Pular

00
Respostas

▲ b = a

◆ a = b

● aux = b

■ a = aux

kahoot.it PIN do jogo: **1278848**

O valor completo da variável **n** para o resultado imprimido será **987654321** .

11

```
s = '123456789'
n = #...
print(s[:n])
```

Pular

00
Respostas

▲ -1

◆ 1

● 2

■ 00

kahoot.it PIN do jogo: **1278848**

Para calcular o fatorial de um número **n** , qual comando está faltando?

26

```
f = 1
for i in range(1, n+1):
    #...
print(f)
```

Pular

00
Respostas

▲ f* = f

◆ f* = f + i

● f* = i

■ f* = f * i

kahoot.it PIN do jogo: **1278848**

Complete o algoritmo para que seja retornado os divisores de um número n .

55

```
n = int(input())
div = []
for i in range(1, n+1):
    if #...:
        div.append(i)
print(div)
```

Pular

00 Respostas

▲ $i \% n == 0$

◆ $n \% i == 0$

● $n \% i! = 0$

■ $i \% n! = 0$

kahoot.it PIN do jogo: 1278848

QUESTÕES DA DIMENSÃO COMPATIBILIZAR

Qual é o código para escrever algo na tela de exibição?

23



Pular

00 Respostas

▲ ('

◆ impressão

● Olá Mundo!

■ ')

kahoot.it PIN do jogo: 6198217

Ordene um código que mostre a idade, dado o ano atual e o ano de nascimento, respectivamente.

53



Pular

00 Respostas

▲ idade = ano - anonasc

◆ impressão (idade)

● anonasc = int (input ('Digite o ano que você nasceu:'))

■ anoatual = int (input ('Digite o ano atual:'))

kahoot.it PIN do jogo: 6198217

Compatibilidade ou código que demonstra números positivos e negativos.



84

Kahoot!

Pular

00 Respostas

▲ elif num! = 0:	◆ print ("negativo")	● se num > 0:	■ print ("positivo")
---------------------	-------------------------	------------------	-------------------------

kahoot.it PIN do jogo: 6198217

Ordene um Programa para Verificar O Maior NUMERO between two Números Diferentes n_1 e n_2 .



55

Kahoot!

Pular

00 Respostas

▲ impressão (n2)	◆ outro:	● se $n_1 > n_2$:	■ impressão (n1)
---------------------	-------------	-----------------------	---------------------

kahoot.it PIN do jogo: 6198217

Qual é a ordem deste laço de repetição?



54

Kahoot!

Pular

00 Respostas

▲ no intervalo	◆ x	● (0, 5):	■ para
-------------------	--------	--------------	-----------

kahoot.it PIN do jogo: 6198217

Compatibilize um laço para percorrer e multiplicar por 2 os elementos de uma lista x .

57

Kahoot!

Pular

00 Respostas

▲ impressão (x)	◆ para i no intervalo (0, 4):	● $x[i] * = 2$	■ $x = [1, 2, 3, 4]$
--------------------	----------------------------------	-------------------	-------------------------

kahoot.it PIN do jogo: 6198217

Ordene o Programa Para Mostrar um Multiplicação de Dois Números um e b POR Meio de APENAS somas.

56.

Kahoot!

Pular

00 Respostas

▲ resultado += a	◆ resultado = 0	● print (resultado)	■ para i na faixa (b):
---------------------	--------------------	------------------------	---------------------------

kahoot.it PIN do jogo: 6198217

Organize o código de modo que esteja escrito na tela o número 5.

56

Kahoot!

Pular

00 Respostas

▲ enquanto eu <5:	◆ impressão (i)	● $i = 0$	■ $i += 1$
----------------------	--------------------	--------------	---------------

kahoot.it PIN do jogo: 6198217

Compatibilidade ou código que calcula e imprime uma tabuada de 10, indo de 1 até 10.

57

Kahoot!

Pular

00 Respostas

▲ i = 1	◆ enquanto i <= 10:	● print ("% d *% d =% d"% (i, n, i * n))	■ i + = 1
------------	------------------------	--	--------------

kahoot.it PIN do jogo: 6198217

Organize o código para adicionar o valor em uma lista e imprimir uma média a cada valor adicionado.

55

Kahoot!

Pular

00 Respostas

▲ L.append (int (input ("Digite os valores:"))))	◆ impressão (soma (L) / i)	● para i no intervalo (1, 5):	■ L = []
---	----------------------------------	-------------------------------------	-------------

kahoot.it PIN do jogo: 6198217

QUESTÕES DA DIMENSÃO CORRIGIR

Goku gostaria de escrever sua frase favorita: **read (oi, eu sou o goku)** . Qual o erro do trecho?

25

Kahoot!

Pular

00 Respostas

▲ Escrito com aspas simples, deve ser duplas.	◆ Escrito para ler ao ativar a impressão .
● Escrito parênteses ao usar colchões.	■ Escrito sem ponto e vírgula.

kahoot.it PIN do jogo: 7612320

O que causa diferença na compilação do seu código, pode causar erros ou soluções incorretas?



21



Pular

00

Respostas

▲ Número de aspas (simples ou duplas) na hora de imprimir.

◆ Uso do ponto e vírgula.

● Comentar com # (por linha) ou "" "" (trecho de código).

■ Indentação do código.

kahoot.it PIN do jogo: 7612320

Qual das seguintes atribuições está correta?



23



Pular

00

Respostas

▲ `Zidade = int (input ("Digite sua idade:"))`

◆ `média = (nota1 + nota2 + nota3) / 3`

● `var2 = 1 + 2 - (3 * 4)`

■ `lista = [2, 3, 5, 7, 11]`

kahoot.it PIN do jogo: 7612320

Enzo precisava saber se não havia como passar o ano. O que exige correção no programa?



86

```

algoritmo "exemplo1"
var
  n1, n2, soma: real

inicio

escreva("Digite um número: ")
leia(n1)
escreva("Digite outro número: ")
leia(n2)
escreva("Primeiro número = ", n1)
escreva("Segundo número = ", n2)
escreva("Soma = ", soma)

```

Pular

00

Respostas

▲ Uma variável deve ser do tipo **inteiro** .

◆ Notas e médias são variáveis de tipos diferentes.

● O algoritmo tem início, mas não tem fim.

■ A soma deve ser feita antes do usuário inserir os dados.

kahoot.it PIN do jogo: 7612320

O diretor da UFPA pediu sua ajuda com um código para imprimir uma mensagem. Qual é o erro nele?



53

```

1 turno = input('Digite o seu turno: ')
2 if turno == 'M':
3     print("Bom dia!")
4
5     elif turno == 'V':
6         print('Boa tarde!')
7
8     elif turno == 'N':
9         print("Boa noite!")
10
11 else:
12     print('Valor inválido!')
```

Pular

00
Respostas

▲ Erro de sintaxe (depois do **resto**, deve ter uma condição).

◆ Erro de entrada (deve ser **int (entrada ())** ao usar **entrada ()**).

● Erro de string (strings devem estar todas com aspas duplas).

■ Erro de identificação (**se estiver** na mesma coluna).

kahoot.it PIN do jogo: 7612320

Claudomiro desgaste imprimir todos os números de 1 a 99, mas não conseguiu. Onde está o erro?



54

```

1 -for i in range(1, 100):
2     L = []
3     if i % 2 == 0:
4         L.append(i)
5 -print(L)
```

Pular

00
Respostas

▲ **i % 2** deve ser igual a 1 na linha 3.

◆ **L.append (i)** deve ser trocado por **L + = [i]** na linha 4.

● As linhas 1 e 2 devem ser trocadas de lugar.

■ **L.append (i)** deve ser trocado por **L = L.append (i)** na linha 4.

kahoot.it PIN do jogo: 7612320

Nenhum código a seguir, o que está impedindo o programa de rodar corretamente?



54

```

1 letra = input("esta chovendo? (S/N) \n")
2
3 if letra == "S":
4     chuva = True
5 elif letra == "N":
6     chuva = False
7
8 if chuva:
9     print(está chovendo)
10 else:
11     print(nao esta chovendo)
12
```

Pular

00
Respostas

▲ Deveriam ter aspas nas linhas 9 e 11.

◆ Deveria ser **chuva == Verdadeiro** na linha 8.

● Deveria ser **int (input (...))** na linha 1.

■ Deveria ser **==** nas linhas 4 e 6.

kahoot.it PIN do jogo: 7612320

Ao Tubsclaison rodar este código no Python, ele encontrou um erro. O que deveria ser feito?



57

```
1 nota1 = input()
2 nota2 = input()
3 media = (nota1 + nota2)/2
4 -print(media)
```

Pular

00
Respostas

▲ Deveria ser **print ("media")** na linha 4.

◆ Deveria ser **float (input ())** nas linhas 1 e 2.

● Deveria ter **media = 0** antes da linha 1.

■ Deveria ser **media += (nota1 + nota2) / 2** na linha 3.

kahoot.it PIN do jogo: 7612320

Este código está errado. O que deve ser feito para corrigir?



53

```
1 n1 = int(input())
2 n2 = int(input())
3 n3 = int(input())
4
5 if N1 > N2 and N1 > N3:
6     print("O maior é igual a {}".format(N1))
7 -elif N2 > N1 and N2 > N3:
8     print("O maior é igual a {}".format(N2))
9 -elif N3 > N1 and N3 > N2:
10    print("O maior é igual a {}".format(N3))
```

Pular

00
Respostas

▲ Deveria ser **ou** ao usar de **e** nas linhas 5, 7 e 9.

◆ Deveria ser **else** na linha 9.

● Deveria ser **N1 , N2 e N3** nas linhas 1, 2 e 3.

■ Deveria ser **<** ao usar de **>** nas linhas 5, 7 e 9.

kahoot.it PIN do jogo: 7612320

Um código onde é solicitado pelo usuário (**admin**) e a senha (**123**) não está funcionando. Onde está o erro?



86

```
1 usuario = input("Digite o nome do usuário: ")
2 senha = input("Digite a sua senha: ")
3
4 while usuario != "admin" or senha != "123":
5     if usuario != "admin" and senha == "123":
6         print("Senha e senha está incorreta! Tente novamente.")
7     elif usuario == "admin" and senha != "123":
8         print("Senha e usuário está incorreto! Tente novamente.")
9     elif usuario == senha:
10        print("O usuário não pode ser igual à senha!")
11 else:
12     print("Ambos usuário e senha estão incorretos.")
13
14 usuario = input("Digite o nome do usuário: ")
15 senha = input("Digite a sua senha: ")
16
17 print("Você entrou no sistema!")
```

Pular

00
Respostas

▲ Na linha 9, deve ser o maior **usuário! = Senha** .

◆ Na linha 4, deve ser **e** ao usar de **ou** .

● As linhas 6 e 8 estão invertidas.

■ Ocorre loop infinito no **while** da linha 4 (roda para sempre).

kahoot.it PIN do jogo: 7612320