



**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**

**UM FRAMEWORK GAMIFICADO PARA A DISCIPLINA ALGORITMOS OU
EQUIVALENTE**

PLANO DE ENSINO

**DISCENTE: JOSÉ AUGUSTO DE SENA QUARESMA
ORIENTADORA: PROFA. DRA. MARIANNE KOGUT ELIASQUEVICI
COORIENTADOR: PROF. DR. SANDRO RONALDO BEZERRA OLIVEIRA**

**BELÉM
2019**



APRESENTAÇÃO

Este Plano de Ensino é parte integrante do *Framework Gamificado*, produto educacional, fruto da pesquisa de mestrado do discente José Augusto de Sena Quaresma, sob orientação da professora Doutora Marianne Kogut Eliasquevici e coorientação do professor Doutor Sandro Ronaldo Bezerra Oliveira, no Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior (PPGCIMES), do Núcleo de Inovação e Tecnologias Aplicadas a Ensino e Extensão (NITAE²), da Universidade Federal do Pará (UFPA). A dissertação de mestrado intitula-se: “Um *framework gamificado* para a disciplina algoritmos ou equivalente”.

Para uso dos(as) docentes, o produto encontra-se disponível acessando os *links* a seguir:

- Plano de Ensino: <http://twixar.me/w811>
- Planilha *Gamificada*: <http://twixar.me/b811>

Tanto o Plano de Ensino, quanto a Planilha *Gamificada*, podem ser adaptadas, desde que citadas as fontes de referência.

Esperamos que a experiência seja enriquecedora para todos(as)!

<<Nome da Universidade>>

<<Nome da Faculdade>>

<<
Logo
Instituição
>>

PLANO DE ENSINO-APRENDIZAGEM *GAMIFICADO*¹

1 IDENTIFICAÇÃO

Professor(a): <<Nome do professor(a)>>

Disciplina: <<Nome da disciplina. Ex: Algoritmos>>

Curso: <<Nome do Curso>>

Semestre: <<Semestre de aplicação>>

Ano: <<Ano de aplicação>>

Carga horária: <<Quantidade de horas geral, semanal e encontros semanais>>

2 CONTEÚDO

- Estrutura de dados homogêneas:
 - Vetor
 - Matriz

3 OBJETIVO GERAL DA DISCIPLINA

Desenvolver no aluno a capacidade de compreender e elaborar algoritmos com qualidade e eficiência para a solução de problemas computacionais, independente de uma linguagem de programação.

3.1 OBJETIVO DE ENSINO PARA ESTA UNIDADE DE CONTEÚDO

Capacitar o discente da Disciplina após a conclusão dos estudos sobre o conteúdo, em desenvolver e analisar Estrutura de dados homogênea.

3.2 OBJETIVOS DE APRENDIZAGEM PARA ESTA UNIDADE DE CONTEÚDO

¹ Ao usar este Plano de Ensino, é importante citar que foi adaptado desta dissertação de mestrado defendida no Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior.

1. Identificar estrutura de dados homogêneos em algoritmos.
2. Implementar estrutura de dados homogêneos em algoritmos e na linguagem de programação adotada para a disciplina.
3. Realizar teste para analisar qual o resultado do processamento da estrutura de dados homogêneos.

3.3 HABILIDADES E COMPETÊNCIAS (ESPECÍFICAS)

Especificar, modelar, implantar e avaliar estrutura de dados homogêneas em algoritmos.

4 PLANEJAMENTO

Para o conteúdo de que trata esse plano de ensino, foi adotado um *framework gamificado*, adaptado de Elgrably e Oliveira (2018) e Santos e Oliveira (2018). As informações referentes ao processo de *gamificação*, bem como as atividades, os recursos metodológicos e a forma como serão avaliados os alunos está presente na seção 5 desse plano de ensino.

5 PROCESSO DE GAMIFICAÇÃO

A *gamificação* é definida segundo Deterding *et al.* (2011) como o uso de elementos de jogos em ambientes que não são de jogos. A partir dessa premissa pretendemos *gamificar* as atividades em sala de aula ao se trabalhar com algumas dinâmicas, mecânicas e elementos de games para ensinar o conteúdo de estrutura de dados homogêneos.

Portanto, a avaliação e planejamento do processo de ensino e aprendizagem no conteúdo de estrutura de dados homogêneos serão baseados na *gamificação* aqui descrita. De maneira inicial informamos que os alunos serão avaliados de duas maneiras: uma em relação ao conteúdo, ao qual ele receberá pontos e um *avatar* a cada aula. E, a outra em relação a comportamento, ao qual o aluno receberá estrelas a cada aula.

5.1 JOGADORES

Nessa seção será descrito os personagens participantes dessa *gamificação*, a qual está descrito no Quadro 1.

Quadro 1 – Descrição de papéis

Identificação	Função	Nome
Master	Professor/ Pesquisador	<<Nome>>
Mentora	Professora da Disciplina	<<Nome>>
Coach	Acompanhamento na resolução de problemas	<<Nome do monitor>>
Juiz/ Scoreplay	Colaborador para desenvolvimento do score da turma	<<Nome>>
Players	Jogadores/ estudantes	Alunos da disciplina Algoritmos

5.2 DESCRIÇÃO DAS ATIVIDADES NO JOGO COM O CRONOGRAMA

Para efeito dessa *gamificação* identificamos como os dias de aula, atividades práticas desenvolvidas e a prova como abordagens. A informação referente a data, máxima pontuação e estrela que podem ser alcançadas, bem como uma rápida descrição de cada abordagem, estão apresentadas no Quadro 2.

O local onde acontecerão as aulas e atividades do *framework gamificado* será o <<Identifique a localização do laboratório ou sala de aula>>, localizado no <<Instituto e/ou Universidade aonde está localizado>>.

Os recursos que estarão disponíveis para essa *gamificação* será a estrutura do laboratório, o uso de um notebook pessoal do pesquisador e pinceis atômicos.

Quadro 2 – Cronograma das fases e a pontuação atribuída

Data	Abordagem	Atividade	Pontuação	Quantidade de Estrelas
dd/mm/aaaa	Conversa inicial sobre a <i>gamificação</i>	Não terá atividades	0	0
dd/mm/aaaa	Aula teórica e prática – Array Unidimensional (Seção 5.2.1)	Lista exercício sala – 4 questões	100	3
dd/mm/aaaa	Aula teórica e prática – Array Bidimensional	Lista exercício sala – 4 questões	100	3
dd/mm/aaaa	Dojo Randori (Seção 5.2.3)	Atividade em dupla rotativa avaliando a narrativa (copiloto) e a execução (piloto)	200	3
dd/mm/aaaa	Dojo Randori	Atividade em dupla rotativa avaliando a narrativa (copiloto) e a execução (piloto)		3
dd/mm/aaaa	Atividade Lab (Seção 5.2.4)	Atividade em dupla fixa, resolvendo 4 questões. Lista de exercício de casa (4 questões)	300 (200 – Atividade em sala + 100 – Lista de exercício de casa)	0
dd/mm/aaaa	Prova Teórica (Seção 5.2.5)	Teste individual (4 questões)	300	0

Data	Abordagem	Atividade	Pontuação	Quantidade de Estrelas
dd/mm/aaaa	Feedback (Seção (5.2.6))	Bonificação sobre a análise do ensino de <i>gamificação</i>		3

5.2.1 Aula Teórica e Prática

A aula teórica e prática é uma abordagem híbrida de ensino em que o professor mescla conceitos teóricos com práticas de programação. Para essa abordagem foram reservados dois dias de aula, no primeiro dia será focado em vetor e no segundo dia os estudos serão voltados para matriz.

Como trata-se de um processo *gamificado* o aluno ganhará pontuação e estrelas no dia, sendo que o processo de ganho de estrela está descrito na seção 5.5 desse documento. O aluno pontuará ao resolver as questões do exercício proposto. Para essa pontuação, utilizaremos a seguinte regra: O aluno ganha 10 pontos pela resolução da primeira questão, 20 pela segunda, 30 pela terceira e 40 pela quarta, por possuírem níveis de dificuldades diferentes; A lista de exercício proposta por dia possui quatro questões (Apêndices A e B), então diariamente o aluno terá um total de 100 pontos; A medida que o aluno receber a pontuação ele receberá um *avatar* diário, segundo a Quadro 3.

A condicional de teste para validar a pontuação do desafio pela resolução do aluno estará unicamente disponível para o professor e monitor da disciplina.

Quadro 3 – Avatar diário Aula Teórica e Prática

Avatar	Faixa de pontuação
Analista	90 – 100
Programador	70 – 89
Aprendiz	50 – 69
Amador	0 – 49

5.2.1.1 Procedimentos Metodológicos

As duas aulas teóricas e práticas terão a mesma estrutura lógica de apresentação do conteúdo, contendo: uma Metáfora sobre o assunto; apresentação do conceito e a forma como é implementado em pascal; um exercício exemplo para o professor fazer junto com os alunos; uma lista de exercício com quatro questões; Por fim, serão retomados os pontos abordados na aula.

5.2.2 Dojo *Randori*

A abordagem Dojo *Randori* é uma prática de ensino de programação em ambiente seguro. O aluno aprenderá por tentativa e erro, com o auxílio de seus pares. Tal prática é adaptada do *framework* apresentado por Luz e Neto (2012). A abordagem possui os seguintes papéis: (i) piloto, responsável pelo desenvolvimento do código; (ii) copiloto, com a função de narrar e auxiliar no planejamento e codificação do programa; (iii) plateia, interage entre si com a intenção de cooperar com o piloto e copiloto durante o desenvolvimento. No contexto da sala de aula será uma das nossas dinâmicas. Contará com dois dias de desenvolvimento por conta do tamanho da turma e cada um dos alunos passará pelos três papéis já descritos.

Na atividade em questão os alunos serão avaliados por comportamento segundo os quesitos definidos na seção 5.5 desse documento. No que tange o conteúdo o aluno receberá pontuação segundo os quesitos estabelecidos na Quadro 4. O quesito pontuações pelas questões desenvolvidas pela turma será uma nota atribuída pelo professor em relação a codificação das funções realizadas pela turma, podendo ir de 0 até 50 pontos, ou seja, nesse atributo toda a turma receberá nota igual. As questões para a atividade do Dojo *Randori* estão presentes no Apêndice D.

Quadro 4 – Pontos adquiridos no Dojo *Randori*

Quesito	Pontos adquiridos
Uma questão concluída	20
Utilizou o fluxo Programar-Testar	30
Narrou como copiloto	100
Pontuações pelas questões desenvolvidas pela turma	Até 50

O *avatar* atribuído para a atividade do Dojo *Randori* seguirá a regra estabelecida na Quadro 5.

Quadro 5 – *Avatar* Dojo *Randori*

<i>Avatar</i>	Faixa de pontuação
Analista	180 – 200
Programador	150 – 179
Aprendiz	120 – 149
Amador	0 – 119

5.2.2.1 Procedimentos Metodológicos

As duas aulas de Dojo seguirão o mesmo procedimento metodológico, o qual será: o professor explicará sobre as regras do Dojo *Randori* e apresentará a lista de exercício proposta

pra que os alunos programem em par; Após esse momento será solicitado que dois voluntários assumam os papéis de piloto e copiloto; Eles iniciam a codificação, após cinco minutos da atividade é permitido a plateia emitir sugestão na programação; Depois de dois minutos o piloto retorna à plateia, o copiloto assume a posição de piloto e um voluntário da plateia assume a posição de copiloto. O ciclo se repete até que todos os alunos tenham participado da atividade. Ao final, o professor recolherá os códigos gerados para análise.

5.2.3 Atividade LAB

A atividade Laboratório de Programação, abreviada como LAB, é uma abordagem proposta no *framework* de Elgrably e Oliveira (2018) e Santos e Oliveira (2018). Na proposta em questão, os alunos serão divididos em duplas cada uma com acesso a um computador, o qual deve estar sem acesso à internet, e terão uma lista com quatro desafios para responder.

Os alunos ganharam, nesse dia, pontos por cada desafio, sendo dividido da seguinte maneira: (i) 20 pontos para o desafio concluído; (ii) 30 pontos atribuídos pelo professor em relação a nota da codificação; (iii) A pontuação máxima por desafio será de 50 pontos. Como cada dupla receberá uma lista contendo quatro questões (Apêndice E) a pontuação máxima para o LAB será de 200 pontos. Além disso, para esse dia, foi reservado a entrega para o exercício feito em casa (Apêndice C), o qual pontuará segundo as regras descritas: (i) 25 pontos por questão resolvida de maneira correta; (ii) Um total de 100 pontos para as quatro questões resolvidas de maneira correta. Sendo assim, a pontuação total do dia será a soma dos pontos adquiridos, tanto na atividade LAB, quanto para a lista de exercício feito em casa. Portanto, para esse dia, teremos o *avatar* diário, segundo condições estabelecidas na Quadro 6. Logo, os alunos não receberão recurso estrela como descrito, na seção 5.5 desse documento.

Quadro 6 – *Avatar* da Atividade LAB em conjunto com a lista de exercício feita em casa

<i>Avatar</i>	Faixa de pontuação
Analista	270 – 300
Programador	230 – 269
Aprendiz	190 – 229
Amador	0 – 189

5.2.3.1 Procedimentos Metodológicos

Na atividade LAB será solicitado aos alunos para que se dividam em duplas e após a divisão serão organizados em um computador para cada dupla. Cada dupla receberá uma folha

com as quatro questões referentes. Após a conclusão o professor recolherá os códigos fontes das duplas para analisar. O tempo para a execução dessa atividade será de 1 hora e 30 minutos.

5.2.4 Teste Teórico

O teste teórico verifica o conhecimento do aluno, individualmente, em relação ao conteúdo estudado na disciplina. Nesse dia os alunos não receberão recurso estrela, segundo descrição da seção 5.5 desse documento. Porém, os alunos pontuarão segundo as seguintes regras para cada questão: (i) 25 pontos pela questão concluída; (ii) Somados a de 0 a 50 pontos atribuídos pela análise do código feito pelo professor. O teste teórico conterá quatro questões (Apêndice F) cada uma com o valor máximo de 75 pontos, portando o nível máximo de pontuação do aluno será de 300 pontos. E, como estamos em um processo *gamificado*, o aluno ganhará um *avatar* pelo seu desempenho na prova segundo regras definidas na Quadro 7.

Quadro 7 – *Avatar* Teste Teórico

<i>Avatar</i>	Faixa de pontuação
Analista	270 – 300
Programador	230 – 269
Aprendiz	190 – 229
Amador	0 – 189

5.2.4.1 Procedimentos Metodológicos

Para o teste teórico, os alunos farão a atividade individualmente e terão o tempo de 1 hora e 30 minutos para resolver quatro questões sobre o conteúdo estrutura de dados homogêneos.

5.2.5 Aula de *Feedback*

A aula de *feedback* é uma abordagem prevista no *framework* de Elgrably e Oliveira (2018) e Santos e Oliveira (2018), em que os alunos serão questionados pelo professor sobre o processo de ensino e aprendizagem via *gamificação*. As perguntas desse dia se encontram no Apêndice G. Nessa ocasião, os alunos ganharam apenas recurso estrela, segundo regras descritas na seção 5.5 desse documento.

5.2.5.1 Procedimentos metodológicos

Na aula de feedback o professor instigará os alunos por meio de uma apresentação em Power point sobre as diversas atividades que aconteceram no *framework gamificado*, com o intuito de melhorar para uma próxima turma.

5.3 Mecânicas de bônus

A bonificação é um recurso em que o aluno ganhará por seu comportamento em sala de aula, após a contabilização de todos os bônus e penalidades que o aluno sofreu, esse recurso será transformado em estrela, segundo as regras descritas na seção 5.5. As regras que definem a bonificação do aluno estão descritas no Quadro 8.

Quadro 8 – Distribuição das bonificações

Nome da dimensão	Explicação da dimensão	Bônus por dimensão	Máxima bonificação por dimensão	Justificativa
Presença	O aluno estar presente pontualmente no horário da aula.	10 Bônus	10 Bônus	A presença é considerada importante para compreender o conteúdo
Participação	0 – Caso o aluno não tenha participado; 2 bônus a cada participação até no máximo 10 bônus de participação (Ex: Feito algum comentário para o professor, respondido a uma pergunta aberta do professor a turma).	2 Bônus	10 Bônus	A participação em sala de aula indica que o aluno está atento ao conteúdo
Sugestão	0 – Caso o aluno não tenha sugerido; 2 bônus a cada sugestão com até no máximo 10 bônus de sugestão.	2 Bônus	10 Bônus	A sugestão mostra que o aluno está buscando o conceito além da sala de aula.
Pergunta	0 – Caso o aluno não tenha perguntado; 2 – bônus a cada pergunta com até no máximo 10 bônus de pergunta	2 Bônus	10 Bônus	A pergunta caracteriza o interesse pelo assunto abordado.
Colaboração	0 – Caso o aluno não tenha ajudado algum colega; 2 bônus a cada ajuda ao colega na resolução das atividades proposta com até no máximo 10 bônus de colaboração.	2 Bônus	10 Bônus	A colaboração nutre no aluno o sentimento de auxílio ao colega.

5.4 Mecânicas de penalidades

A penalidade é uma infração cometida pelo aluno a qual será penalizada com a perda de bonificação. As regras que definem a bonificação do aluno estão descritas no Quadro 9.

Quadro 9 – Distribuição das penalidades

Nome da dimensão	Explicação da dimensão	Perda de bônus por dimensão	Justificativa
Falta	-10 pontos caso falte a disciplina	-10	A falta caracteriza a perda do conteúdo passado no dia.
Atraso em 10 minutos	O aluno perderá 2 bônus a cada atraso. Podendo ter a penalidade aplicada a no máximo 4 atrasos, totalizando perda de 8 bônus.	-2	O atraso em 10 minutos identifica o desinteresse do aluno pela disciplina
Uso do celular	A cada uso do celular em horário de aula o aluno perderá 2 bônus	-2	O uso do celular indica a não atenção a atividade ou dinâmica desenvolvida em sala
Uso do computador	Utilizar o computador em sites e ferramentas não previstas nos procedimentos metodológicos	-2	O uso do computador indica a não atenção a atividade ou dinâmica desenvolvida em sala
Penalidade por atrapalhar a aula	Quando o aluno interromper a aula com brincadeiras ou não permitir que outro aluno assista a aula perderá 2 bônus.	-2	Atrapalhar a aula indica a não atenção a atividade também pode atrapalhar o desempenho dos outros alunos

5.5 Conversão de bônus para estrela

Após os alunos bonificarem e sofrerem as penalidades, diariamente receberam o recurso estrela exceto nos casos já previstos. As regras para concessão do recurso estrela estão descritas no Quadro 10.

Quadro 10 – Conversão de bônus para recurso estrela

Recurso	Descrição	Intervalo de bônus	Quantidade
Estrela	O recurso estrela é atribuído ao jogador conforme a sua bonificação	34 – 50	3
		17 – 34	2
		5 – 17	1

5.6 Conversão da pontuação para nota na disciplina

A pontuação final do jogo ocorrerá mediante a transformação do recurso estrela em pontos, o qual acontece segundo Quadro 11. E, a transformação do processo *gamificado* em nota será o número de pontos dividido por 100.

Quadro 11 – Conversão das estrelas para pontuação

Faixa de pontuação	Nota extra
15 – 10	100
9 – 5	50
4 – 0	0

5.7 Medalhas da *gamificação* e premiação

A concessão de medalha para a *gamificação* acontecerá no final das aulas e poderá ser acompanhada pelos discentes por meio da aba Medalhas. Para a *gamificação* da nossa sala de aula, teremos dois tipos de medalha: Power, referente as pontuações adquiridas no decorrer do processo e a Participativo, em relação ao recurso estrela adquirido. Para cada medalha teremos três posições, são elas: 1º lugar, ouro; 2º lugar, prata; 3º lugar, bronze. O critério de desempate para a medalha Participativo será o número total de bonificações sem a conversão para recurso estrela.

Os primeiros lugares, tanto na medalha Power quanto na medalha Participativo, ganharão o seguinte prêmio: um curso online pela plataforma Udemy.

6 REFERÊNCIA BIBLIOGRÁFICA BÁSICA

CARVALHO, S. E.R. **Introdução à Programação com PASCAL**. 4.ed. Rio de Janeiro: Campus, 1986.

DETERDING, S.; DIXON, D.; KHALED, R.; NACKE, L. From game design elements to gamefulness: defining gamification. In: **Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments**. ACM, 2011. p. 9-15.

EIGRABLY, I. S.; OLIVEIRA, S. R. B. "Gamification and Evaluation of the Use the Agile Tests in Software Quality Subjects: the Application of Experiments". **13th ENASE - Evaluation of Novel Approaches to Software Engineering**. Madeira, Portugal, 2018.

EVARISTO, Jaime. Programando com Pascal. **2ª Edição**. Ed. Book Express, 2004.

LUZ, R. B.; NETO, A. Using Programming Dojos for Test-Driven Development Teaching. **XXIII Brazilian Symposium on Informatics in Education (SBIE 2012)**. Rio de Janeiro. p.25-35, 26, 2012.

SANTOS, E. D.; OLIVEIRA, S. R. B. Gamificação como Ferramenta de Suporte de Apoio ao Ensino da Técnica Análise de Pontos por Função em um Turma de Pós-Graduação: Um Estudo de Caso. **15th CONTECSI - International Conference on Information Systems and Technology Management**. São Paulo, Brasil. 2018.

7 REFERÊNCIA BIBLIOGRÁFICA COMPLEMENTAR

MEDINA, Marco; FERTING, Cristina. **Algoritmos e programação: teoria e prática**. São Paulo: Novatec Editora, 2006.

APÊNDICE A - LISTA DE EXERCÍCIO 01

Conteúdo: Estrutura de dados homogêneos (vetor)

Observação:

Lista disponível para os alunos no dia <<Informe o dia da prática>>

A lista precisa ser feita em aula para os alunos ganharem pontuação

1) Desenvolva um algoritmo que leia do usuário um vetor de 10 posições e calcule e exiba o número armazenado mais o índice da posição que ele está armazenado.

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5 6 7 8 9 10

O programa deve apresentar: 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10

2) Desenvolva um algoritmo em pascal que leia um vetor de 7 posições e calcule e exiba primeiro os números armazenados com índice ímpar e depois em índice par.

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5 6 7

O programa deve apresentar: 1 3 5 7

2 4 6

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

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5

6 7 8 9 10

O programa deve apresentar: 7 9 11 13 15

4) Desenvolva um algoritmo em pascal que leia os 12 meses do ano e, à medida que o usuário digitar um número, informe o mês correspondente; quando o usuário digitar 0, finalize o programa, caso digite outro número informe mês inválido.

Condição para concessão da pontuação:

Inserir no programa: 1/ 3 /5/ 0

O programa deve apresentar: Janeiro/ Março/ Maio

APÊNDICE B - LISTA DE EXERCÍCIO 02

Conteúdo: Estrutura de dados homogêneos (matriz)

Observação:

Lista disponível para os alunos no dia <<Informe o dia da prática>>

A lista precisa ser feita em aula para os alunos ganharem pontuação

1) 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.

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5 6 7 8 9

O programa deve apresentar: 1 2 3 4 5 6 7 8 9

2) Desenvolva um algoritmo que calcule e exiba o traço e a diagonal da matriz informada na primeira questão. A regra do cálculo do traço da matriz é a soma dos elementos da diagonal principal dela. A diagonal principal é formada pelos elementos cujo índice de colunas é igual ao índice de linhas.

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5 6 7 8 9

O programa deve apresentar: 1 5 9

3) Desenvolva um algoritmo para armazenar e exibir a agenda do usuário, portanto o programa deve receber os seguintes dados pessoais dos 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 e o usuário informará na primeira interação todos os contatos.

Condição para concessão da pontuação:

Inserir no programa: José Rua 66073070 Guamá 9198888888/ Maria Travessa 66073070 São Braz 9197777777/ João Rua 66073070 Benguí 9193333333/ Jorge Avenida 66073070 Doca 9190000000/ Rebeca Travessa 66073070 Guamá 919355678 /Ezequiel Rua 66073070 Guamá 919252852/ Ester Rua 66073070 Marco 9198888888

O programa deve apresentar: José Rua 66073070 Guamá 9198888888/ Maria Travessa 66073070 São Braz 9197777777/ João Rua 66073070 Benguí 9193333333/ Jorge Avenida 66073070 Doca 9190000000/ Rebeca Travessa 66073070 Guamá 919355678/ Ezequiel Rua 66073070 Guamá 919252852/ Ester Rua 66073070 Marco 9198888888

4) 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.

Condição para concessão da pontuação:

Inserir no programa: 1 2/ 3 4/ 5 6/ 1 2/ 3 4/ 4 5/ 4 5/ 3 4/ 3 3/ 1 2/

O programa deve apresentar: 1 2 2 / 3 4 12/ 5 6 30/1 2 2/ 3 4 12/ 4 5 20/ 4 5 20/ 3 4 12/ 3 3 9/
1 2 2

APÊNDICE C – LISTA DE EXERCÍCIO PARA CASA

Conteúdo: Estrutura de dados homogêneos

Observação:

Lista disponível para os alunos no dia <<Informe o dia da prática>>

A lista será feita em casa para os alunos ganharem pontuação

1) Desenvolva um programa em Pascal que receba do usuário um vetor de 10 posições, multiplique os valores em índice ímpar por 10 e divida os valores em índice par por 10. Depois apresente ao usuário o vetor resultante.

2) Desenvolva um programa em Pascal que leia do usuário dois vetores de 5 posições do tipo inteiro, multiplique o primeiro pelo inverso do segundo e apresente um terceiro vetor com o resultado para o usuário.

3) Elabore um programa em Pascal que leia dois vetores com 5 elementos do tipo inteiro, cada. Intercale os elementos dos dois vetores, formando um novo vetor com 10 elementos.

EX:

6	1	8
---	---	---

3	4	7
---	---	---

6	3	1	4	8	7
---	---	---	---	---	---

4) Criar um programa em Pascal que, dado um vetor de 5 elementos do tipo inteiro, crie um outro que seja o inverso do primeiro e apresente os dois vetores para o usuário.

(Ex.: vet1 → 1 2 3 4 5; vet2 → 5 4 3 2 1)

5) Escrever um programa em Pascal que lê um vetor N com 20 elementos do tipo inteiro e o escreve. Em seguida, troque o 1º elemento com o último, o 2º com o penúltimo até o 10º com o 11º e escreva o vetor N assim modificado.

6) Desenvolva um programa em Pascal que implemente uma matriz 3x3, receba do usuário números reais, coloque os números na matriz e depois apresente para o usuário apenas os elementos da diagonal principal.

7) Elabore um programa em Pascal que crie uma matriz 7X8 em que cada elemento é a soma dos índices de sua posição dentro da matriz. Ao final, apresente para o usuário a matriz gerada em formato matricial.

- 8) Criar um programa em Pascal que possa armazenar as alturas de dez atletas de cinco delegações que participarão dos jogos de verão. Imprimir a maior altura de cada delegação.
- 9) Elabore um algoritmo em pascal que receba números inteiros informados pelo usuário, armazene em uma matriz 3x3 e identifique se existem números repetidos, quantas vezes se repetem e apresente ao usuário.
- 10) Faça um programa que leia uma matriz 10X10, calcule e escreva a soma dos elementos que estão na área marcada com '*'.

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

APÊNDICE D – LISTA DE EXERCÍCIOS PARA O DOJO *RANDORI*

Conteúdo: Estrutura de dados homogêneos

Observação:

Lista disponível para os alunos no dia <<Informe o primeiro dia da prática>>

Atividade 01

A empresa de desenvolvimento Açai Tecnologia foi contratada para desenvolver um programa em Pascal para uma empresa de ônibus. O programa deve ser capaz: de reservar, pagar e cancelar reserva no Ônibus; o assento reservado guarda o valor do RG da pessoa; caso o assento já tenha sido reservado, informar que o assento não está disponível; exibir para o usuário os assentos disponíveis em formato de matriz; exibir os assentos já ocupados; informar se o ônibus já está lotado. Para fins de simplificação considere que o ônibus possui 60 lugares, com fileiras de 4 lugares disponíveis.

Lista de exercício para o segundo dia <<Informe o segundo dia da prática>>

Atividade 01

Uma empresa foi contratada para criar um programa para uma loja. O programa tem que ser capaz de: incluir e excluir as vendas de cada semana dos meses do ano; alterar as vendas de uma determinada semana de um mês específico; exibir o total vendido em cada mês do ano; exibir o total vendido no ano e, ainda, em que mês(es) a loja alcançou o mínimo desejado para vendas (R\$ 5.000,00). Para fins de simplificação, considere que cada mês do ano possui somente 04 semanas

APÊNDICE E – ATIVIDADE LABORATÓRIO DE PROGRAMAÇÃO

Conteúdo: Estrutura de dados homogêneos

Observação:

Atividade disponível para os alunos no dia <<Informe o dia da prática>>

A atividade será realizada em dupla

Não será liberada consulta a internet, aos materiais disponibilizados e nem a dupla ao lado

1) Desenvolva um algoritmo em Pascal que contenha a lista de preços para um leilão *online*. O programa recebe todos os lances e apresenta para o usuário o lance com preço maior, menor e a média dos lances que foram realizados. O vetor receberá 15 lances.

2) Elabore um programa em Pascal que armazene os 40 primeiros números primos em um vetor de inteiros. Depois apresente lado a lado os números de índice ímpar e os números de índice par. Ex:

Ímpar		Par
2		3
5		7

3) Elabore um algoritmo em pascal que leia do usuário números inteiros e os armazene em uma matriz de ordem 4, depois calcule o traço da diagonal principal. Limpe a tela do *prompt* e apresente ao usuário os números que estão acima da diagonal principal e que são maiores que o traço, em formato horizontal separado por “|”.

4) Sua empresa foi contratada por uma rede de cinemas para gerar um programa para reservar as cadeiras para uma sessão. Como bom desenvolvedor verificou que as filas começam com a

numeração 1 e vão até a 12, também observou que em cada fila da sessão tem 12 poltronas, logo, uma matriz de ordem 12.

Desenvolva um algoritmo em pascal que realize a “reserva” de lugares (informando a fileira e a cadeira), com o nome do cliente, caso a cadeira já tenha sido reservada, exibir uma mensagem para o cliente, exibir em formato matricial as cadeiras disponíveis, permitir uma opção para o cliente, verificar se a sessão já está cheia, e ao digitar a palavra “fim” o programa finaliza.

APÊNDICE F – TESTE TEÓRICO

Conteúdo: Estrutura de dados homogêneos

Observação:

Atividade disponível para os alunos no dia <<Informe o dia da prática>>

A atividade será realizada individualmente.

Não será liberada consulta a internet, aos materiais disponibilizados e nem o colega ao lado.

O aluno realizará a atividade no papel com caneta, sem o auxílio de computador.

1) Faça um programa em Pascal que leia um vetor de 5 posições de números inteiros, multiplique todos os elementos pelo maior valor do vetor e apresente o novo vetor.

2) Faça um programa em Pascal que leia um vetor de seis elementos numéricos inteiros, calcule e mostre:

- a) A quantidade de números pares
- b) Quais os números pares
- c) A quantidade de números ímpares
- d) Quais os números ímpares

3) Desenvolva um algoritmo em Pascal que leia os valores de uma matriz de ordem 3 de inteiros, calcule e apresente a média dos números digitados, assim como a matriz.

4) Desenvolva um algoritmo em Pascal que leia os valores de uma matriz de ordem 4 e apresente todos os elementos da diagonal secundária, bem como a soma deles.

APÊNDICE G – ATIVIDADE FEEDBACK

Data da atividade: <<Informe o dia da prática>>

As questões que serão feitas a turma para análise do *framework gamificado* foram divididas em categorias e seguirão a ordem abaixo:

Feedback sobre as metodologias de ensino escolhidas:

- Foram positivas para o aprendizado? Por quê?
- Seria necessário mais tempo de aula teórica e prática? Por quê?
- Necessita de mais tempo na aula de Dojo *Randori*? Por quê?
- Qual a melhor forma do Dojo na sua opinião Tipo 01 ou Tipo 02? Por quê?
- Necessitaria de mais tempo na aula prática LAB (Laboratório de Programação)? Por quê?
- Você acha que seria necessário algum pré-requisito para um melhor aproveitamento dos alunos nas metodologias escolhidas para o ensino? Quais?

Feedback sobre a aula teórica e prática:

- A aula expositiva e prática foi satisfatória para o aprendizado de Estrutura de dados homogêneos?
- Qual dificuldade vocês acharam para as listas de exercício?
- Vocês consideram válida a experiência do fluxo de ensino:
- Metáfora -> conceito -> A implementação na linguagem Pascal -> Um exercício exemplo
- O fluxo de ensino foi satisfatório ou gostariam de outra forma de ensino?

Feedback sobre a aula de Dojo *Randori*:

- A prática do Dojo foi considerada satisfatória?
- Existia a necessidade de mais tempo para a explicação sobre as regras e o funcionamento do Dojo?
- Qual dificuldade vocês acharam da atividade?
- Vocês acham que o tempo destinado para atividade foi apropriado? Ou queriam mais tempo dessa atividade?

- Vocês têm alguma sugestão para melhoria da prática do Dojo?

Feedback sobre a prática de LAB:

- A prática do LAB foi considerada satisfatória?
- O tempo que foi disponibilizado para a prática foi suficiente?
- Qual a dificuldade encontrada na atividade?
- Fazer a técnica utilizando pareamento, foi positivo?
- Vocês têm alguma sugestão para melhoria da prática do LAB?

Feedback sobre a lista de exercício feita em casa:

- Você achou interessante ter uma lista de exercício para ser desenvolvida em casa?
- Qual a dificuldade vocês acham para a lista de exercício?

Feedback sobre a importância de Estrutura de dados homogêneos:

- Houve interesse em se aprofundar em estrutura de dados homogêneos?
- A utilização de Dojo e LAB, facilitou o entendimento da importância da prática da implementação e análise de Estrutura de dados homogêneos?
- Houve interesse dos alunos em utilizar a estrutura de dados homogêneos em projetos pessoais, no estágio ou no emprego?