MAUÁ	PREENCHA OS CAMPOS EM AMARELO	PROVA P4
Disc.:	ECM306 – TÓPICOS AVANÇADOS EM ESTRUTURA DE DADOS	
Curso:	ENGENHARIA DA COMPUTAÇÃO	

Aluno: **Pedro Wilian Palumbo Bevilacqua**		*Test Code*: **CJZSB**
Curso: **ECM** Série **3ª** Período: **Matutino**		
São Caetano do Sul, 13 de novembro de 2025.	RA: **23.01307-9**	
Assinatura: **Pedro Wilian Palumbo Bevilacqua**	Nota:	

Instruções da Prova

1. Esta prova é individual e prática, devendo ser realizada nos computadores do IMT, sendo permitido ao aluno, se assim desejar, utilizar seu próprio computador, sob sua inteira responsabilidade;
2. Não poderá haver acesso à Internet, sob nenhuma circunstância, exceto ao CanvasLMS do próprio aluno e, mesmo assim, somente em duas etapas: para receber ("baixar") as questões de sua prova na máquina em uso; e para entregar ("subir") as resoluções das questões de sua prova no devido local de entrega, na mesma plataforma.

   IMPORTANTE: O aluno deverá informar ao professor quando fará os dois acessos permitidos a ele ao CanvasLMS, pela Internet;
3. Poderá haver consulta a qualquer material do próprio aluno, seja ele físico (livros, artigos, material de aula etc.) ou virtual (livros, artigos, material de aula, exercícios, resoluções etc.), desde que esse material esteja previamente armazenado em seu computador ou em qualquer dispositivo de armazenamento externo (*pendrive*, *hd* externo etc.).

   IMPORTANTE: Não será permitido o acesso pela Internet a pastas compartilhadas (*Google Drive*, *OneDrive* etc.), nem a repositórios virtuais (*GitHub*, *GitLab*, *BitBucket* etc.), mesmo sendo de posse e administração do próprio aluno;
4. O aluno deverá responder às questões da prova no próprio arquivo da prova (*.docx*) devidamente identificado (RA e Nome completo do Aluno) e quando for o caso, gerando os códigos necessários, somente na linguagem de programação JAVA e utilizando sempre e somente o paradigma da Programação Orientada à Objetos visto nas aulas;
5. Para realizar a entrega da prova na plataforma CanvasLMS, o aluno deverá gerar e entregar um único arquivo compactado (*.rar* ou *.zip*), tendo seu RA e seu NOME completo como nome desse arquivo. Nesse arquivo compactado o aluno deverá fornecer, obrigatoriamente, os seus dados acadêmicos preenchidos na prova, bem como a resolução das questões nela solicitadas (*.docx*), além dos arquivos e códigos gerados em suas resoluções, uma pasta para cada questão, contendo as suas classes e demais arquivos que possibilitem sua posterior execução pelo professor, durante a resolução.

   IMPORTANTE: Na correção, para executar o código gerado pelo aluno em sua prova, o professor seguirá exatamente as instruções fornecidas pelos alunos e contidas nas resoluções das provas! Caso não obtenha sucesso, a questão será considerada errada.
6. Não poderá haver troca de informações, nem de materiais, sejam físicos ou virtuais, entre os alunos durante a prova;
7. Não é permitido ao estudante se ausentar da sala antes da entrega da prova;
8. Celulares e outros equipamentos eletrônicos (exceto o notebook do aluno, se assim optar) devem permanecer desligados enquanto o estudante estiver na sala;
9. O tempo limite para realização da prova é de **90** minutos;
10. Mantenha sobre a carteira apenas um documento com foto, caneta, lápis e borracha;
11. O entendimento das questões faz parte da avaliação;
12. O tempo mínimo de permanência na sala é de **30** minutos;
13. O estudante que chegar atrasado em até **30** minutos do início da prova poderá fazê-la.

Questão 1: (*1 ponto*)

Segundo os autores *M.C. Nicoletti* e *E.R. Hruschka Jr.* em *Fundamentos da Teoria dos Grafos para Computação* (3ª edição, 2017), um emparelhamento em um grafo *G* é um conjunto de arestas duas a duas não adjacentes. Um emparelhamento é dito perfeito quando todos os vértices de *G* são incidentes a alguma aresta desse conjunto.

Com base na bibliografia e no material estudado em aula, avalie as afirmações a seguir:

I. *Um emparelhamento perfeito só pode existir em grafos bipartidos.*

II. *Se um grafo possui um emparelhamento perfeito, o número total de vértices deve ser par.*

III. *Todo emparelhamento perfeito é, necessariamente, máximo.*

IV. *Se um grafo bipartido possui um emparelhamento máximo, ele também possui um emparelhamento perfeito.*

É correto, apenas, o que se afirma em:

a) I e II;
b) II e III;
c) I e III;
d) I e IV;
e) III e IV.

<u>Questão 2</u>: (*1 ponto*)

O Teorema de *Berge* afirma que um emparelhamento *M* é máximo se, e somente se, não existir um caminho *aumentante* em relação a *M*. Caminhos *aumentantes* e *alternantes* desempenham papel fundamental no estudo de emparelhamentos.

Sendo assim, com base na referida bibliografia e no material de aula utilizados, avalie as afirmações a seguir:

I. *Um caminho aumentante alterna arestas pertencentes e não pertencentes ao emparelhamento M.*

II. *A existência de um caminho aumentante garante que é possível obter um emparelhamento maior que M.*

III. *A ausência de caminhos aumentantes garante que M é um emparelhamento perfeito.*

IV. *Se um emparelhamento é máximo, então nenhum caminho aumentante existe em relação a ele.*

É correto, apenas, o que se afirma em:

a) I e III;
b) II e III;
c) I, II e IV;
d) III e IV;
e) II e IV.

Questão 3: (*1 ponto*)

Segundo *Nicoletti & Hruschka Jr*. (*2017*), um dos critérios clássicos de *Hamiltonianidade* é o Teorema de *Dirac*, o qual estabelece:

"Se *G* é um grafo simples com $n \geq 3$ vértices e o grau de cada vértice satisfaz $d(G) \geq n/2$, então *G* é *Hamiltoniano*."

Considere um grafo *G* com 8 vértices, definido pelos conjuntos:

- V = {1, 2, 3, 4, 5, 6, 7, 8};
- As arestas conectam cada vértice aos seus quatro vizinhos mais próximos em um ciclo:
  - 1 liga-se a 2, 3, 8, 7;
  - 2 liga-se a 1, 3, 4, 8;
  - 3 liga-se a 1, 2, 4, 5;
  - 4 liga-se a 2, 3, 5, 6;
  - 5 liga-se a 3, 4, 6, 7;
  - 6 liga-se a 4, 5, 7, 8;
  - 7 liga-se a 1, 5, 6, 8;
  - 8 liga-se a 1, 2, 6, 7.

Levando-se em consideração esse contexto, a bibliografia adotada e o material de aula utilizado, avalie as seguintes asserções e a relação proposta entre elas:

I. *Como o grafo G possui* $n = 8$ *vértices e cada vértice tem grau 4, então G satisfaz as condições impostas pelo Teorema de Dirac.*

PORQUE

II. *Pelo Teorema de Dirac, o grafo G é Hamiltoniano, pois atende a condição de grau mínimo necessária.*

A respeito dessas asserções, assinale a única opção correta:

a) As asserções I e II são proposições falsas;

b) As asserções I e II são proposições verdadeiras e a II é uma justificativa da I;

c) As asserções I e II são proposições verdadeiras e a I é uma justificativa da II;

d) A asserção I é uma proposição falsa e a II é verdadeira;

e) A asserção I é verdadeira e a II é falsa.

Questão 4: (*1 ponto*)

Segundo os autores M.C. Nicoletti e E.R. Hruschka em seu livro "Fundamentos da Teoria dos Grafos para Computação", 3ª edição – LTC, os **grafos** são estruturas matemáticas utilizadas para representar relações entre objetos. Um grafo é composto por Vértices (ou nós) e Arestas (ou arcos). Dentre os diversos tipos de grafos, temos os seguintes tipos com propriedades específicas: Grafo Euleriano, Grafo Semi-euleriano, Grafo Hamiltoniano e Grafo Semi-hamiltoniano.

Para uma determinada situação, é dado o grafo G, composto pelos vértices e arestas abaixo:

Vértices: V = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

Arestas: E = {(1, 2), (2, 3), (3, 4), (4, 5), (5, 6), (6, 7), (7, 8), (8, 9), (9, 10), (10, 1), (1, 5), (1, 6), (5, 10), (6, 9), (2, 7)}

Levando-se em consideração esse contexto, a bibliografia adotada e o material de aula utilizado, avalie as seguintes asserções e a relação proposta entre elas:

I. Pelo teorema de Euler, um grafo conexo é Euleriano se e somente se todos vértices tiverem grau par.

PORQUE

II. O grafo G não é Euleriano.

A respeito dessas asserções, assinale a única opção correta:

a) As asserções I e II são proposições falsas;

b) As asserções I e II são proposições verdadeiras e a II é uma justificativa da I;

c) As asserções I e II são proposições verdadeiras e a I é uma justificativa da II;

d) A asserção I é uma proposição falsa e a II é uma proposição verdadeira;

e) A asserção I é uma proposição verdadeira e a II é uma proposição falsa.

Questão 5: (*1 ponto*)

Segundo o Teorema da Curva de *Jordan*, apresentado originalmente por *Camille Jordan* (1893), toda curva simples e fechada no plano — isto é, uma curva contínua que não se auto intersecta — divide o plano em exatamente duas regiões disjuntas: uma região interior, limitada pela curva, e uma região exterior, ilimitada. Esse resultado é fundamental em diversas áreas da computação, como computação gráfica, geometria computacional e análise de colisões, pois garante a separação do plano em duas partes bem definidas, mesmo quando a curva tem formato irregular.

Considere agora uma curva simples e fechada C definida no plano cartesiano, de forma que sua projeção delimita uma região interna composta por todos os pontos (x, y) que satisfazem uma relação contínua de contorno, sem auto intersecções, e uma região externa composta por todos os demais pontos.

Com base no Teorema da Curva de *Jordan* e nos conceitos apresentados e discutidos em aula, assinale a alternativa que melhor descreve a consequência lógica da aplicação desse teorema à curva C:

a) A curva C pode delimitar múltiplas regiões internas desde que o contorno seja contínuo e fechado.

b) A curva C só divide o plano em duas regiões se for uma curva convexa.

c) A curva C pode dividir o plano em mais de duas regiões desde que possua trechos diferenciáveis.

d) A curva C só garante duas regiões distintas se for definida por funções polinomiais contínuas.

e) A curva C necessariamente separa o plano em uma região interior e uma exterior, ambas conexas e disjuntas, independentemente do formato da curva, desde que ela seja simples e fechada.

Questão 6: (5 *pontos*)

Gerado por qualquer *IDE* fornecer o código fonte, em *Java*, das classes necessárias (incluindo a de execução), implementadas sob o paradigma da *Programação Orientada a Objetos*, além do programa ***.jar*** que as executa diretamente, objetivando resolver o problema descrito a seguir.

ATENÇÃO:

i. Se o programa *.jar* fornecido não executar automaticamente a aplicação desenvolvida, a resolução da questão será invalidada (0 ponto);

ii. As classes soluções desta questão (arquivos *.java*) deverão ser compactadas em um único arquivo (*.zip* ou *.rar*), em conjunto com o respectivo arquivo *.jar* funcional, além deste arquivo *.docx* da prova, contendo todas as respostas para as outras questões e a devida identificação do aluno e entregue em resposta à tarefa do *CanvasLMS* da prova.

Problema:

Utilizando apenas os códigos vistos e praticados em aula e os códigos desenvolvidos pelo próprio aluno para esta disciplina, através de uma aplicação implementada durante a prova, e utilizando os conceitos apresentados pela Teoria dos Grafos, deseja-se resolver o seguinte problema:

"Em uma clínica, através de um sistema computacional *online*, pacientes precisarão agendar consultas com os médicos que lá trabalham. Num determinado dia e horário (por exemplo, na próxima segunda-feira, às 9h), 5 (cinco) pacientes precisam escolher 1 (um) entre 6 (seis) médicos disponíveis de sua preferência. Cada paciente poderá escolher mais de um médico como opção alternativa".

Elabore um programa que receba, via digitação, cada um dos pacientes (*P1*, *P2*, *P3*, *P4* ou *P5*) a serem consultados e com qual(is) médico(s) (*M1*, *M2*, *M3*, *M4*, *M5* e *M6*) cada um desses pacientes deseja se consultar.

O sistema deverá acomodar a(s) preferência(s) de cada um dos 5 (cinco) pacientes para esse determinado dia e horário em uma **Estrutura de Dados GRAFO baseada em nós (alocação dinâmica)**.

A seguir, partindo do GRAFO gerado e preenchido, o sistema deverá apresentar em tela a situação geral dos agendamentos (**SEM RESOLVER O PROBLEMA DE EMPARELHAMENTO**).

Como praticante da técnica de *TDD* já estudada anteriormente, a seguir estão definidos os casos de testes que deverão guiar o programador na elaboração do programa solicitado:

Casos de Teste:

CASO DE TESTE 1:

Entradas	**Digitação**
P1:	*M1, M2*
P2:	*M1, M3*
P3:	*M3*
P4:	*M4*
P5:	*M6*

Saída

Clínica Tabajara
Agenda de Alocação de Consultas
Dia: *Próxima 2a feira, 9h00min*

Médicos:	**M1**	**M2**	**M3**	**M4**	**M5**	**M6**
Pacientes:	*P1, P2*	*P1*	*P2, P3*	*P4*	-	*P5*

CASO DE TESTE 2:

Entradas	**Digitação**
P1:	*M2*
P2:	*M1, M3*
P3:	*M3*
P4:	*M3, M5*
P5:	*M5*

Saída

Clínica Tabajara
Agenda de Alocação de Consultas
Dia: *Próxima 2a feira, 9h00min*

Médicos:	**M1**	**M2**	**M3**	**M4**	**M5**	**M6**
Pacientes:	*P2*	*P1*	*P2, P3, P4*	-	*P4, P5*	-