

ATIVIDADE 3 - ESTRUTURAS DE DADOS - 52_2025**Período:**02/06/2025 08:00 a 06/07/2025 23:59 (Horário de Brasília)**Status:**ABERTO**Nota máxima:**0,50**Gabarito:**Gabarito será liberado no dia 07/07/2025 00:00 (Horário de Brasília)**Nota obtida:****1ª QUESTÃO**

Filas são amplamente empregadas como estruturas de dados, embora sua dinâmica apresente complexidades adicionais em comparação com pilhas. O princípio fundamental subjacente a todas as filas é o FIFO (First In, First Out), que, traduzido, significa que o primeiro elemento a ser inserido na fila é também o primeiro a ser retirado dela. Assim sendo, analise o trecho de código a seguir contendo a estrutura de dados da fila:

```
typedef struct {  
    int itens[MAX]; // MAX é o tamanho máximo da fila  
    int frente, tras;  
} Fila;
```

Fonte: Elaborado pelo professor, 2024.

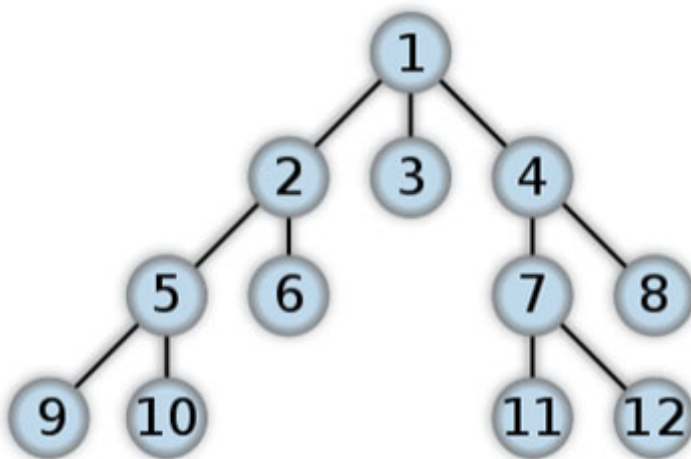
Assinale a alternativa que contenha o trecho de código que faça a implementação correta da função em C para verificar se uma fila está vazia.

ALTERNATIVAS

```
int fila_vazia(Fila *f) {  
    if (f->frente == f->tras)  
        return 1;  
    else  
        return 0;  
}  
int fila_vazia(Fila *f) {  
    if (f->frente == -1 && f->tras == -1)  
        return 1;  
    else  
        return 0;  
}  
int fila_vazia(Fila *f) {  
    if (f->frente == 0 && f->tras == 0)  
        return 1;  
    else  
        return 0;  
}  
int fila_vazia(Fila *f) {  
    if (f->frente == -1 || f->tras == -1)  
        return 1;  
    else  
        return 0;  
}  
int fila_vazia(Fila *f) {  
    if (f->frente == NULL && f->tras == NULL)  
        return 1;  
    else  
        return 0;  
}
```

2ª QUESTÃO

Observe o grafo abaixo. Considerando que todos os vértices, à esquerda do vértice 4, já foram visitados. Qual o caminho percorrido, ao se realizar a busca em profundidade (interna), começando do vértice 4 em direção ao vértice 8?



Com base no exposto, assinale a alternativa correta.

ALTERNATIVAS

- ☐ 4, 8.
- ☐ 4, 7, 8.
- ☐ 4, 7, 11, 8.
- ☐ 4, 8, 7, 12, 11.
- ☒ 4, 7, 11, 12, 8.

3ª QUESTÃO

Em computação, a pilha é um tipo de estrutura na qual os dados são adicionados e removidos exclusivamente do topo. Essas estruturas são comumente referidas como Last In, First Out (LIFO), o que significa que o último dado a ser inserido será o primeiro a ser removido.

Fonte: Elaborado pelo professor, 2024.

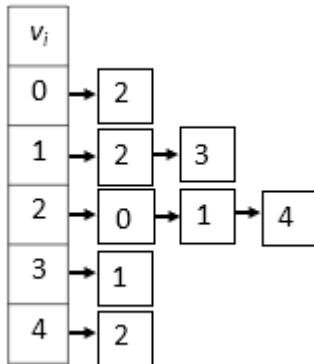
Qual é o resultado da operação em uma pilha que tinha vários elementos, mas no momento da execução de desempilhar um elemento a pilha estava vazia?

ALTERNATIVAS

- ☐ Retorna NULL.
- ☒ Deveria lançar uma exceção de pilha vazia.
- ☐ Retorna o elemento que foi removido da pilha.
- ☐ Remove o elemento da pilha sem retornar qualquer valor.
- ☐ Aumenta o tamanho da pilha para 1 e insere o elemento desempilhado.

4ª QUESTÃO

Existem bastante opções para um(a) programador implementar algoritmos em grafos. As opções dependem, na realidade, muitas vezes, da forma com a qual um grafo é representado enquanto estruturas de dados. Sabendo disso, observe a ilustração a seguir:



Com base em seus conhecimentos sobre representação computacional de grafos, é possível dizer que a figura acima implementa um grafo por meio de:

ALTERNATIVAS

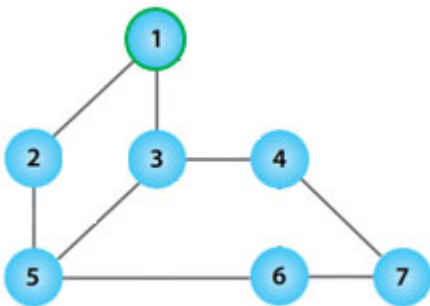
- ☐ Uma matriz de adjacências.
- ☐ Uma matriz de arestas.
- ☐ Uma pilha de vértices.
- ☐ Um vetor de arcos.
- ☒ Uma lista de adjacências.

5ª QUESTÃO

A busca em largura de baseia no conceito de fila, no qual para cada nó que está no início da fila, é preciso visitar todos os seus adjacentes, antes de verificar os adjacentes de um novo nó, de maneira recorrente. Observe o algoritmo BFS(), abaixo:

1. **BFS**(Grafo G, Vértice início):
2. Marque ***início*** como visitado, e enfileire-o
3. Para cada **vértice *v*** adjacente ao ***corrente***
4. Marque ***v*** como visitado, enfileire ***v***
5. Desenfileire
6. Se a ***fila*** não está vazia
7. Volte ao passo **3**
8. Se não encerre o algoritmo

Assim sendo, aplique o algoritmo acima no grafo representado abaixo, considerando que a busca se inicia no vértice 1.



A sequência de visitação que corresponde à resposta correta é:

ALTERNATIVAS

- ☐ 1, 2, 5, 3, 4, 7, 6.
- ☐ 1, 3, 2, 5, 4, 7, 6.
- ☐ 1, 3, 4, 7, 6, 5, 2.
- ☒ 1, 2, 3, 5, 4, 6, 7.
- ☐ 2, 3, 4, 5, 6, 7.

6ª QUESTÃO

"Muitos problemas podem ser descritos por meio de grafos, nos quais a solução para o problema requer que realizemos uma busca pelo grafo. As buscas, em geral, partem de um nó inicial em direção a um nó alvo, fazendo com que tenhamos que percorrer toda uma sequência ordenada de nós e arestas. Além disso, o próprio caminho, em si, pode ser objeto da busca, isto é, às vezes a solução reside no caminho percorrido, e não em um nó alvo específico."

OLIVEIRA, Pietro Martins de; PEREIRA, Rogério de Leon. Estruturas de Dados I. Maringá: Unicesumar, 2019.

Considerando tanto o algoritmo de busca em largura, quanto em profundidade, para que seja possível que tais algoritmos consigam navegar por todos os nós de um grafo, é imprescindível que o grafo seja:

ALTERNATIVAS

- ☐ Um dígrafo.
- ☐ Orientado.
- ☒ Conexo.
- ☐ Ponderado.
- ☐ Um multigrafo.

7ª QUESTÃO

Um grafo é uma estrutura $G = (V, E)$, em que V é um conjunto finito de vértices (ou nós) e E um conjunto de arestas (ou arcos). Cada arco, em um grafo, é especificado por um par de nós. Diante dessa explicação, assinale a alternativa que demonstra como é chamado um grafo, quando especificamente seus arcos forem formados por pares ordenados de vértices:

ALTERNATIVAS

- ☐ Trivial.
- ☐ Conexo.
- ☒ Orientado.
- ☐ Desconexo.
- ☐ Desorientado.

8ª QUESTÃO

Considere o seguinte cenário:

Você está visitando um médico para um check-up. Há muitas pessoas na clínica. Uma senhora está digitando os nomes de todas as pessoas em um arquivo. A pessoa que chegar mais cedo será atendida primeiro. Quando o médico está livre, ele chama o próximo paciente na sequência de chegada.

Assim, com base nesta informação, avalie as asserções a seguir e a relação proposta entre elas:

I. O cenário acima representa a estrutura Fila porque segue o método primeiro que entra é o primeiro que sai.

PORQUE

II. Em uma fila adicionamos elementos na parte de trás e os removemos na frente.

A respeito dessas asserções, assinale a opção correta.

ALTERNATIVAS

- ☒ As asserções I e II são proposições verdadeiras e a II é uma justificativa correta da I.
- ☐ As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa correta da I.
- ☐ A asserção I é uma proposição verdadeira e a II é uma proposição falsa.
- ☐ A asserção I é uma proposição falsa e a II é uma proposição verdadeira.
- ☐ As asserções I e II são proposições falsas.

9ª QUESTÃO

O trecho de código abaixo tem o intuito de implementar a função de inserção em uma fila:

OLIVEIRA, Pietro Martins de; PEREIRA, Rogério de Leon. **Estruturas de Dados I**. Maringá: Unicesumar, 2019.

Linha	Código
	#define tamanho 3
	struct tfila {
01	int dados
02	
03	<i>tamanho</i>
04	;
05	int fim;
06	};
07	
08	struct tfila fila;
09	
10	void insere(){
11	if (fila.fim == tamanho) {
12	printf("\nA fila está cheia, volte outro dia!\n\n");
13	system("pause");
14	}
15	else {
16	printf("\nDigite o valor a ser inserido: ");
17	scanf("%d", &fila.dados
18	
19	<i>fila.fim</i>
20);
	fila.fim++;
	}
	}

Com base no código-fonte acima, e em seus conhecimentos sobre a implementação de filas estáticas, avalie as afirmações que se seguem.

I – Da forma como foi implementada essa fila, ao tentar inserir um elemento na posição de índice igual a 3, no vetor de dados, teremos uma situação de erro.

II – A função de inserção de dados na fila não recebe argumento algum, e depende de interação com o usuário.

III – A função de inserção está, obrigatoriamente, incompleta, já que não atualiza o índice de início da fila.

De acordo com as afirmações acima, é possível dizer que está(ão) correta(s) a(s) afirmativa(s):

ALTERNATIVAS

- ☐ I, apenas.
- ☐ II, apenas.
- ☐ III, apenas.
- ☒ I e II, apenas.
- ☐ II e III, apenas.

10ª QUESTÃO

Ao trabalharmos com vetores em linguagem de programação, temos duas opções de uso: estáticos e dinâmicos. Quando criamos um vetor estático, dizemos na sua declaração qual será o seu tamanho, mas em vetores dinâmicos o seu tamanho definido em tempo de execução do programa.

Fonte: Elaborado pelo professor, 2024.

Qual é a principal vantagem dos vetores dinâmicos em comparação com os vetores estáticos em linguagens de programação?

ALTERNATIVAS

- ☐ Vetores dinâmicos possuem tamanho fixo, o que facilita a alocação de memória.
 - ☐ Vetores dinâmicos são mais eficientes em termos de uso de memória e tempo de execução do que os vetores estáticos.
 - ☐ Vetores dinâmicos só podem armazenar um tipo específico de dados, enquanto os vetores estáticos podem armazenar uma variedade de tipos.
 - ☐ Vetores dinâmicos são estáticos em termos de tamanho, mas dinâmicos em termos de conteúdo, o que os torna mais previsíveis e seguros.
 - ☒ Vetores dinâmicos permitem redimensionamento durante a execução do programa, adaptando-se às necessidades de armazenamento de dados.
-