Calendário Lives

Menu das Semanas

Notas

Semana 1

Semana 2

Semana 3

Semana 4

Semana 5

Semana 6

Semana 7

Semana 8

Avaliação

Documentos e

Gabaritos

informações gerais

Facilitadores da disciplina

Repositório de REA's

Página Inicial1

Exame

Orientações Gerais para a

```
Revisar envio do teste: Semana 5 - Atividade Avaliativa
Tempo decorrido
                  14 minutos
Resultados exibidos Todas as respostas, Respostas enviadas, Respostas corretas, Comentários, Perguntas respondidas incorretamente
   Pergunta 1
                                                                                                                                     1,43 em 1,43 pontos
            O algoritmo de ordenação Quick Sort escolhe um pivô que corresponde ao primeiro elemento da lista e o troca de posição com o elemento
            do meio da lista. É iniciada a varredura da lista comparando os elementos com esse pivô, de forma que os elementos _____
            ele são colocados ou mantidos na lista do lado esquerdo, e os elementos _____ que ele são colocados ou mantidos na lista do
            lado direito. Ao realizar esse processo de forma ______, chega-se ao final com uma lista totalmente ordenada.
            Preencha as lacunas escolhendo a alternativa correta.
             Resposta Selecionada: o h. menores — maiores — recursiva.
                                   a maiores — iguais — iterativa.
             Respostas:
                                 <sub>b.</sub> menores — maiores — recursiva.
                                    c menores — maiores — iterativa.
                                    d. maiores — iguais — recursiva.
                                    <sub>e</sub> maiores — menores — iterativa.
             Comentário
                          JUSTIFICATIVA
             da resposta:
                          A primeira lacuna é completada pelo termo "menores", pois, no algoritmo de ordenação Quick Sort, escolhe-se um pivô, que é
                          o primeiro elemento da lista, para que seja trocado de posição com o elemento que se encontra no meio da lista. A lista
                          começa a ser percorrida verificando cada elemento e comparando-o com o pivô escolhido, se o elemento que está sendo
                          comparado estiver em uma posição menor do que o pivô pela ordem alfabética, ele é transferido ou mantido na lista da
                          de que todo o processo acontece de forma recursiva, produzindo, ao final, uma lista ordenada.
   Pergunta 2
            Considere o detalhamento a seguir em relação a um algoritmo de ordenação que se baseia em comparação local:
            1. Se o elemento for o primeiro, ele já encontra-se classificado;
            2. É feita a escolha do próximo elemento;
            3. Ele é comparado com os elementos na sublista classificada inicialmente;
            4. São movidos os elementos na sublista classificada que são maiores que o elemento a ser ordenado;
            5. O elemento é inserido;
             6. O processo de 1 a 5 é repetido até a sublista classificada ser toda a lista.
            citados.
```

```
esquerda. A segunda lacuna é completada pelo termo "maiores", porque, durante o processo de percorrer a lista realizando a
                      comparação com o pivô, se o elemento a ser comparado estiver em uma posição maior do que a do pivô pela ordem
                      alfabética, ele é movido ou mantido na lista da direita. A terceira lacuna é completada pelo termo "recursiva", devido ao fato
                                                                                                                              1,43 em 1,43 pontos
        Analise as alternativas e indique aquela que contém o algoritmo de ordenação cujo processo de ordenação corresponde aos passos
         Resposta Selecionada: e. Insertion Sort.
                               a. Bubble Sort.
         Respostas:
                               b. Heap Sort.
                               <sub>C.</sub> Merge Sort.
                               d. Quick Sort.
                            👩 e. Insertion Sort.
         Comentário
                     JUSTIFICATIVA
         da resposta:
                     O processo citado corresponde ao Insertion Sort, que é um algoritmo de classificação baseado em comparação local em que
                     um subconjunto do vetor mantém-se ordenado e, dessa forma, vai crescendo conforme o vetor vai sendo varrido e cada
                     elemento é devidamente posicionado. A ordenação por inserção faz com que, nesse posicionamento, o elemento novo
                     encontre seu devido lugar para ser inserido. É incorreto citar Bubble Sort porque ele atua com operações em que cada
                     elemento de uma determinada posição (i) é comparado com o elemento ao seu lado (i+1), caso o da posição i seja maior que
                     o da posição i+1, ocorre a troca de lugar até que o maior elemento esteja na última posição. É incorreto citar Quick Sort como
                     o algoritmo do enunciado, porque ele elege um elemento da lista como sendo o pivô e começa a organizar os elementos de
                     modo que os elementos anteriores ao pivô sejam menores e, os à direita do pivô, sejam maiores que ele. É incorreto citar
                     Merge Sort como o algoritmo do enunciado, porque trata-se de um algoritmo eficiente de ordenação por divisão e conquista. É
                     incorreto citar Heap Sort como o algoritmo do enunciado, porque ele utiliza uma estrutura de dados chamada heap binário
                     para ordenar os elementos à medida que os insere na estrutura.
Pergunta 3
                                                                                                                              1,42 em 1,42 pontos
   O algoritmo de ordenação Merge Sort é um dos mais eficientes, dividindo de forma repetitiva uma lista em sublistas, até que reste somente
        um elemento em cada uma dessas sublistas. Após isso, ele começa a fundir essas sublistas e acaba produzindo a lista inicial, porém com
         seus elementos organizados.
         Com base nas informações apresentadas, identifique se são (V) verdadeiras ou (F) falsas as afirmativas a seguir.
         I. ( ) O Merge Sort toma como princípio de funcionamento a divisão e a conquista.
            ) O Merge Sort aplica o merge somente uma vez para conseguir ordenar um vetor.
        III. ( ) Não é realizado o merge de dois vetores, mas sim o merge de duas partes ordenadas em um vetor.
        IV. ( ) O merge é a rotina que agrega dois vetores ordenados em um terceiro não ordenado.
         Assinale a alternativa que apresenta a sequência correta.
         Resposta Selecionada: oa. V, F, V, F.
                            o a. V, F, V, F.
         Respostas:
                               b. V, F, F, F.
                               c. F, F, V, V.
                               d. F, V, V, V.
                               e. V, V, F, F.
         Comentário
                      JUSTIFICATIVA
         da resposta:
                      A afirmativa I é verdadeira, pois o Merge Sort é um algoritmo "dividir-para-conquistar", porque divide os elementos em
                      subsequências pequenas, classifica as duas metades recursivamente, aplicando o Merge Sort e depois une essas duas
                      metades em um único vetor já classificado. A afirmativa II é falsa, porque o Merge Sort aplica o merge inúmeras vezes para
                      conseguir ordenar um vetor, sendo necessárias sucessivas execuções de merge para ordenar um vetor. A afirmativa III é
                      verdadeira, visto que no Merge Sort organiza os elementos no vetor a ser ordenado, de forma que uma parte dele esteja
                      ordenada e outra também, portanto não se faz o merge de dois arrays, mas sim o merge de duas partes ordenadas de um
                      mesmo array. A afirmativa IV é falsa, já que o merge é a rotina que agrega dois vetores ordenados em um único vetor
                      ordenado.
Pergunta 4
                                                                                                                              1,42 em 1,42 pontos
    🖳 Há um algoritmo eficiente para encontrar um elemento presente em uma lista ordenada que, repetidas vezes, separa a parte da lista que
         contém o elemento, a fim de reduzir as possíveis localizações a somente uma localização, sendo assim, a _____
         palpite da localização do elemento procurado que sempre é o elemento localizado no ______ do vetor, caso o palpite seja correto,
        significa que o elemento foi encontrado, mas se o palpite for errado então o próximo palpite fica restrito a uma parte do vetor porque ele
         encontra-se _____.
         Preencha as lacunas escolhendo a alternativa correta.
         Resposta Selecionada: od. busca binária — meio — ordenado.
         Respostas:
                               a. busca linear — fim — ordenado.
                               b. busca binária — meio — desordenado.
                               c. busca binária — fim — ordenado.
                             👩 d. busca binária — meio — ordenado.
                               e busca linear — meio — ordenado.
         Comentário
                      JUSTIFICATIVA
         da resposta:
                      A primeira lacuna é completada pelo termo "busca binária". A busca binária começa examinando o elemento do meio do vetor
                      e, caso esse elemento seja o que está sendo procurado, a busca finaliza, mas, caso não seja, tira-se vantagem do vetor ser
                      ordenado, eliminando metade dele. A segunda lacuna é completada pelo termo "meio", porque, se o elemento procurado for
                      maior que o elemento do meio, deduz-se que o elemento procurado encontra-se na parte superior, portanto a metade inferior
                      do vetor, começando a contar pelo meio do vetor, não necessita ser averiguada. A terceira lacuna é completada pelo termo
                      "ordenado", porque, se o elemento procurado for maior que o elemento do meio, deduz-se que o elemento procurado
```

```
I. Sua utilização é adequada nos casos em que existem informações adicionais sobre os elementos que se deseja pesquisar.
II. A busca linear finaliza ao encontrar o elemento pesquisado (como a[i] == x) ou ao não encontrar o elemento pesquisado.
III. A busca linear compara se a chave de busca é igual ao elemento posicionado no meio da lista e retorna para a posição.
IV. Ao chegar ao final do vetor, há: i == N (x não foi encontrado) e a[i] == x (i é a posição em que x foi localizado).
```

identificador único. O objetivo é verificar se há algum elemento nessa coleção que tenha a chave de busca fornecida.

encontra-se na parte superior, isso indica que a metade restante (inferior) não precisa ser verificada porque já encontra-se

Os algoritmos de busca são aplicados em problemas em que existe uma chave de busca e uma coleção de elementos que têm um

1,42 em 1,42 pontos

Está correto que se afirma em: Resposta Selecionada: _{C. II} e IV, apenas. a. I, II e IV, apenas. Respostas:

b. l e III, apenas.

encontrado na posição i do vetor.

Com relação à busca linear, observe as afirmações a seguir.

```
d. III e IV, apenas.
                      e. I, II e III, apenas.
Comentário
            JUSTIFICATIVA
da resposta:
            A afirmativa I é incorreta, pois a busca linear é adequada em situações em que não existem informações adicionais sobre os
            elementos que se deseja pesquisar, pois consegue varrer a lista fazendo a comparação da chave com os elementos em cada
            uma das posições que ocupam. A afirmativa II é correta, já que, após varrer a lista, se a chave for igual a algum dos
            elementos, é retornada à posição correspondente na lista, mas se toda lista foi varrida, e a chave não foi encontrada, então é
            retornado o valor -1. A afirmativa III é incorreta porque a busca linear compara a chave de busca com os elementos em cada
```

lista.

Respostas:

ordenada.

Pergunta 5

Pergunta 6 1,44 em 1,44 pontos Na intenção de mostrar para os alunos a importância da ordenação interna, um professor apresentou o seguinte conceito: a ordenação de elementos fundamenta-se em sua organização de forma crescente ou decrescente, a fim de facilitar a pesquisa desses elementos, portanto a ordenação foca em facilitar buscas por um elemento que são realizadas em um determinado conjunto de dados. Desse modo, o algoritmo de ordenação deve ser escolhido considerando o tempo utilizado pela ordenação. Após a explicação, um aluno questiona: a escolha do algoritmo de ordenação interna deve basear-se no número de elementos, e não no tempo que a ordenação leva.

posição do vetor. A afirmativa IV é correta, visto que, ao chegar no último elemento do vetor, acontece i == N (significa que o

elemento x não foi encontrado), mas se for a[i] == x (i é a posição em que x foi localizado). E se i != N significa que x foi

PORQUE II. Na existência de uma grande quantidade de elementos a serem ordenados, eles não se acomodam na memória principal, e o acesso a esses elementos ocorre de forma sequencial ou em grandes blocos.

I. O aluno está certo, a escolha pelo algoritmo de ordenação interna deve tomar como base a quantidade de elementos que compõem a

```
A respeito dessas asserções, assinale a alternativa correta.
 Resposta Selecionada: C. As asserções I e II são falsas.
```

🕜 c. As asserções I e II são falsas.

Após análise da situação apresentada, avalie as asserções a seguir e a relação proposta entre elas.

```
e. A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.
Comentário
            JUSTIFICATIVA
da resposta:
            A asserção I é falsa, pois o aluno erra ao afirmar que a escolha pelo algoritmo de ordenação interna deve tomar como base a
            quantidade de elementos que compõem a lista. Portanto, a escolha do algoritmo de ordenação deve considerar o tempo
            gasto pela ordenação, o que indica que se n é o número de elementos constantes na lista, os aspectos importantes
            relacionados à complexidade são a quantidade de comparações entre as chaves e a quantidade de movimentações de
            elementos na lista. A asserção II é falsa, visto que, na ordenação interna, todos os elementos a serem ordenados acomodam-
            se na memória principal, e qualquer elemento pode ser imediatamente acessado, destacando que a utilização econômica da
```

a. As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa da I.

b. As asserções I e II são proposições verdadeiras, e a II é uma justificativa da I.

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

Pergunta 7

falsas.

😱 A pesquisa em memória primária tem a capacidade de encontrar a informação (que é dividida em registros contendo uma chave) desejada em um grande volume de dados. A busca por essa informação requer a escolha de um método de busca que considere a quantidade de dados envolvidos e a periodicidade das operações de inserção e remoção. Considerando a pesquisa em memória primária, avalie as afirmações a seguir em relação aos métodos de pesquisa e as relacione adequadamente aos termos a que se referem. 1. Pesquisa sequencial.

memória disponível é um requisito primordial na ordenação interna. A asserção II não justifica a primeira, pois ambas são

```
I. Adota o paradigma dividir para conquistar, fazendo com que o tempo de busca seja reduzido, pois, a cada iteração do algoritmo, o
tamanho do vetor é dividido ao meio.
```

3. Transformação de chave (hashing).

2. Pesquisa binária.

II. Oferece uma regra de cálculo que possibilita informar o agrupamento para buscar pelos elementos que têm a chave conhecida. III. O elemento procurado é identificado com a pesquisa iniciando no primeiro elemento percorrendo o vetor linearmente, até encontrar a chave procurada.

Resposta Selecionada: d. 1-III; 2-I; 3-II. a. 1-l; 2-lll; 3-ll. Respostas:

```
b. 1-II; 2-I; 3-III.
c. 1-III; 2-II; 3-I.
```

👩 d. 1-III; 2-I; 3-II.

e. 1-l; 2-ll; 3-lll.

Assinale a alternativa que relaciona adequadamente os dois grupos de informações.

```
Comentário
           JUSTIFICATIVA
da resposta:
           1-III. Pesquisa seguencial: os elementos localizam-se em um vetor e são inseridos no final dele. A remoção de algum
           elemento é realizada com a localização dele e, ao ser retirado, em seu lugar é colocado o último elemento do vetor. O
           elemento procurado é identificado com a pesquisa iniciando no primeiro elemento percorrendo o vetor linearmente até
```

encontrar a chave procurada. 2-I. Pesquisa binária: adota o paradigma dividir para conquistar e requer que o vetor esteja ordenado para possibilitar que ele seja dividido em duas partes e verificado em qual dessas partes localiza-se o elemento com a chave procurada. Isso faz com que o tempo de busca seja reduzido, pois, a cada iteração do algoritmo, o tamanho do vetor é dividido ao meio. 3-II. Transformação de chave: considerando um espaço de dados que podem ser rotulados por chave, é possível criar uma forma de dividir tal espaço em agrupamentos baseados em alguma qualidade do domínio das chaves, e isso exige que se saiba quantos agrupamentos devem ser criados; para isso, cria-se uma regra de cálculo que, de posse de uma chave, consiga informar o agrupamento para buscar pelos elementos que contenham tal chave. Esse processo leva o nome de função de hashing.

Quinta-feira, 15 de Agosto de 2024 19h30min27s BRT

 \leftarrow OK

1,44 em 1,44 pontos