Questão 1/10 - Programação III

Considere o trecho de código abaixo:

a = 1

b = a\*10

if a > b:

    c = a

else:

    c = b

 Acerca do código acima são feitas as seguintes afirmativas:

 I. Cada operação (linha de código) tem complexidade assintótica O(1).  
II. Todas as operações são constantes, ou seja, executa da mesma forma para entradas grandes ou pequenas.  
III. A complexidade do código em questão é a soma de todas as complexidades, no caso: O(1) + O(1) + O(1)+ O(1)+ O(1)+ O(1)= O(6n).

Estão corretas as afirmativas:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | I somente. |
|  | B | II somente. |
|  | C | I e II somente.  Você assinalou essa alternativa (C)  **Você acertou!**  **Aula 1 – tema 3** |
|  | D | I e III somente. |
|  | E | II e III somente. |

Questão 2/10 - Programação III

O algoritmo de ordenação pelo método da bolha (bubble sort ) foi utilizado para ordenar a sequência 43, 6, 32, 15, 21 de forma crescente.

Quantas trocas foram realizadas?

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | 2 |
|  | B | 3 |
|  | C | 4  Você assinalou essa alternativa (C) |
|  | D | 5 |
|  | E | 6  **Aula 2 – tema 1** |

Questão 3/10 - Programação III

Na aula 3 de nossa disciplina estudamos estruturas de dados que se comportam como uma FILA. (Adaptada)

Acerca de FILAS, assinale a alternativa CORRETA:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Uma fila onde o primeiro elemento é o 66, o segundo é o 33 e o terceiro é o 99. Inserir na fila significaria inserir um elemento que aponte para o valor 66. |
|  | B | Em uma fila, podemos ter a inserção dos dados no início desta fila. |
|  | C | Em uma fila, podemos ter a remoção dos dados final ou no meio desta fila. |
|  | D | Em uma fila trabalhamos com o conceito de: “o primeiro que entra é o primeiro que sai”.  Você assinalou essa alternativa (D)  **Você acertou!**  Aula 3 – tema 4 |
|  | E | Uma fila onde o primeiro elemento é o 66, o segundo é o 33 e o terceiro é o 99. Remover da fila significaria remover o elemento 99.  Removeria o 66 (remoção no início da fila). |

Questão 4/10 - Programação III

Quanto a análise de algoritmos assinale a alternativa correta:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Quando se fala algoritmo mais eficiente para resolver um problema, estamos nos referindo ao algoritmo de maior complexidade. |
|  | B | A complexidade de tempo analise a quantidade de instruções do código, não impactando diretamente em seu desempenho. |
|  | C | A quantidade de variáveis em um algoritmo e seus tamanhos impactam diretamente em seu desempenho.  Você assinalou essa alternativa (C)  **Você acertou!**  **Aula 1 – tema 2** |
|  | D | Quando um algoritmo  é  complexo em tempo,  ele é necessariamente complexo também em espaço. |
|  | E | A análise de algoritmos é altamente dependentemente do *hardware*. |

Questão 5/10 - Programação III

Uma estrutura de dados operando como uma fila, opera com o princípio de *o primeiro que entra é o primeiro que sai*, ou em inglês, *first in first out* (*fifo*) .

Implementar uma fila significa fazer uma inserção (*queue*) no final dela, e fazer a remoção (*dequeue*)no início dela.

Após realizar a sequencia de operações

QUEUE (11),QUEUE (34) ,DEQUEUE ( ), QUEUE (23) , DEQUEUE ( ) , QUEUE (14) , QUEUE (25) , DEQUEUE ( )

O conteúdo da fila será:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | 25 |
|  | B | 11,34,23,14,25 |
|  | C | 11,23,14,25 |
|  | D | 14,25  **Aula 3 – tema 4** |
|  | E | 11,14  Você assinalou essa alternativa (E) |

Questão 6/10 - Programação III

Considere um vetor ordenado:

O  vetor é dividido ao meio.

O número do meio é comparado com o número procurado. Se forem iguais a busca termina, senão se o número procurado é menor que o do meio, a busca é realizada no subvetor a esquerda, se é maior no subvetor a direita. O procedimento é repetido até que o vetor fique com um elemento ou se encontre o desejado.

 As instruções acima se referem a:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Busca (ou Pesquisa) sequencial |
|  | B | Busca (ou Pesquisa)  Linear |
|  | C | Busca (ou Pesquisa) Binaria  Você assinalou essa alternativa (C)  **Você acertou!**  **Aula 1  - Tema 1** |
|  | D | Ordenação por troca |
|  | E | Ordenação por seleção |

Questão 7/10 - Programação III

A técnica de divisão e conquista consistem de 3 passos básicos:

1.Divisão: Dividir o problema original, em subproblemas menores.   
2.Conquista: Resolver cada subproblema recursivamente.   
3.Combinação: Combinar as soluções encontradas, compondo uma solução para o problema original.

 É um exemplo de algoritmo de divisão e conquista:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Insertion sort |
|  | B | Mergesort  Você assinalou essa alternativa (B)  **Você acertou!**  Aula 2 tema 2 |
|  | C | Bubblesort |
|  | D | Selection sort |
|  | E | Searchsort |

Questão 8/10 - Programação III

**Observe o trecho do algoritmo abaixo e analise o seu comportamento**

X = [6, 5, 2, 3, 4, 1]  
n = 0  
troca = 1  
  
while n <= len(X) and troca == 1:  
    troca = 0  
    for i in range(0, len(X)-1, 1):  
        if X[i] > X[i+1]:  
            troca = 1  
            aux = X[i]  
            X[i] = X[i+1]  
            X[i+1] = aux  
    n = n + 1

Analisando o comportamento do algoritmo que flutua para o topo o maior elemento, pode se afirmar que se trata de qual algoritmo de ordenação?

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | Heapsort  Você assinalou essa alternativa (A) |
|  | B | Mergesort |
|  | C | Quicksort |
|  | D | Bubble sort  Aula 2 tema1 |
|  | E | Insertion sort |

Questão 9/10 - Programação III

Observe a descrição abaixo:

* Comparações são feitas entre elementos de um vetor
* Cada elemento de uma posição i é comparado com o elemento da posição i + 1
* Se a ordenação procurada é encontrada é feita uma troca de posições entre os elementos

A descrição em questão se trata do algoritmo de ordenação:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Mergesort |
|  | B | Quicksort |
|  | C | Bubblesort  Você assinalou essa alternativa (C)  **Você acertou!**  **Aula 2 – tema 1** |
|  | D | Hashsort |
|  | E | Insertion sort |

Questão 10/10 - Programação III

Quanto a busca (pesquisa) em vetores são feitas as seguintes afirmativas:

 I. Na pesquisa sequencial, em um vetor ordenado, deve-se buscar o número até que seja encontrado e enquanto for maior que o número do vetor.

II. Na pesquisa binária, em um vetor não ordenado deve-se buscar o número até que seja encontrado ou até o fim do vetor

III.A pesquisa sequencial é mais eficiente que a pesquisa binária, visto que a pesquisa sequencial funciona tanto par vetores ordenados quanto para vetores desordenados.

Estão corretas as afirmativas:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | I somente.  **Aula 1 – tema 1** |
|  | B | II somente. |
|  | C | I e II somente.  Você assinalou essa alternativa (C) |
|  | D | I e III somente. |
|  | E | II e III somente. |

Questão 1/10 - Programação III

Uma estrutura de dados operando como uma fila, opera com o princípio de *o primeiro que entra é o primeiro que sai*, ou em inglês, *first in first out* (*fifo*) .

Implementar uma fila significa fazer uma inserção (*queue*) no final dela, e fazer a remoção (*dequeue*)no início dela.

Após realizar a sequencia de operações

QUEUE (11),QUEUE (34) ,DEQUEUE ( ), QUEUE (23) , DEQUEUE ( ) , QUEUE (14) , QUEUE (25) , DEQUEUE ( )

O conteúdo da fila será:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | 25 |
|  | B | 11,34,23,14,25 |
|  | C | 11,23,14,25 |
|  | D | 14,25  Você assinalou essa alternativa (D)  **Você acertou!**  **Aula 3 – tema 4** |
|  | E | 11,14 |

Questão 2/10 - Programação III

Considere um vetor ordenado:

O  vetor é dividido ao meio.

O número do meio é comparado com o número procurado. Se forem iguais a busca termina, senão se o número procurado é menor que o do meio, a busca é realizada no subvetor a esquerda, se é maior no subvetor a direita. O procedimento é repetido até que o vetor fique com um elemento ou se encontre o desejado.

 As instruções acima se referem a:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Busca (ou Pesquisa) sequencial |
|  | B | Busca (ou Pesquisa)  Linear |
|  | C | Busca (ou Pesquisa) Binaria  Você assinalou essa alternativa (C)  **Você acertou!**  **Aula 1  - Tema 1** |
|  | D | Ordenação por troca |
|  | E | Ordenação por seleção |

Questão 3/10 - Programação III

**Observe o trecho do algoritmo abaixo e analise o seu comportamento**

X = [6, 5, 2, 3, 4, 1]  
n = 0  
troca = 1  
  
while n <= len(X) and troca == 1:  
    troca = 0  
    for i in range(0, len(X)-1, 1):  
        if X[i] > X[i+1]:  
            troca = 1  
            aux = X[i]  
            X[i] = X[i+1]  
            X[i+1] = aux  
    n = n + 1

Analisando o comportamento do algoritmo que flutua para o topo o maior elemento, pode se afirmar que se trata de qual algoritmo de ordenação?

Nota: 10.0

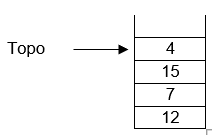
|  |  |  |
| --- | --- | --- |
|  | A | Heapsort |
|  | B | Mergesort |
|  | C | Quicksort |
|  | D | Bubble sort  Você assinalou essa alternativa (D)  **Você acertou!**  Aula 2 tema1 |
|  | E | Insertion sort |

Questão 4/10 - Programação III

Considere o Tipo Abstrato de Dados (TAD) da *Pilha.* Nessa TAD estão implementadas as seguintes operações:

push(x): inserção do elemento x em uma pilha.

pop(): remoção do elemento de uma pilha, retornando esse elemento.

Considere a pilha  abaixo, na qual existem quatro valores armazenados. O topo é indicado pelo ponteiro Topo.  


No estado da pilha acima, foi executada a sequência de instruções abaixo:

y = 0  
x = pop()   
y = y + x  
x = 6   
push(x)  
x = 8  
push(x)  
x = pop()   
x = pop()  
y  = y + x

Ao final da execução o valor de y será:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | 8 |
|  | B | 10  Você assinalou essa alternativa (B)  **Você acertou!**  **Aula 3 – tema 3** |
|  | C | 12 |
|  | D | 19 |
|  | E | 15 |

Questão 5/10 - Programação III

O algoritmo de ordenação pelo método da bolha (bubble sort ) foi utilizado para ordenar a sequência 43, 6, 32, 15, 21 de forma crescente.

Quantas trocas foram realizadas?

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | 2  Você assinalou essa alternativa (A) |
|  | B | 3 |
|  | C | 4 |
|  | D | 5 |
|  | E | 6  **Aula 2 – tema 1** |

Questão 6/10 - Programação III

Na aula 3 de nossa disciplina estudamos estruturas de dados que se comportam como uma FILA. (Adaptada)

Acerca de FILAS, assinale a alternativa CORRETA:

Nota: 10.0

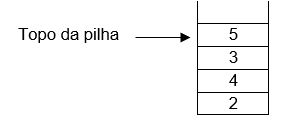
|  |  |  |
| --- | --- | --- |
|  | A | Uma fila onde o primeiro elemento é o 66, o segundo é o 33 e o terceiro é o 99. Inserir na fila significaria inserir um elemento que aponte para o valor 66. |
|  | B | Em uma fila, podemos ter a inserção dos dados no início desta fila. |
|  | C | Em uma fila, podemos ter a remoção dos dados final ou no meio desta fila. |
|  | D | Em uma fila trabalhamos com o conceito de: “o primeiro que entra é o primeiro que sai”.  Você assinalou essa alternativa (D)  **Você acertou!**  Aula 3 – tema 4 |
|  | E | Uma fila onde o primeiro elemento é o 66, o segundo é o 33 e o terceiro é o 99. Remover da fila significaria remover o elemento 99.  Removeria o 66 (remoção no início da fila). |

Questão 7/10 - Programação III

Considere a estrutura de dados do tipo *Pilha* abaixo.  Nessa pilha  existem quatro valores armazenados. O topo é indicado pelo ponteiro Topo da pilha.

A inserção em uma pilha é chamada de *empilhar*, ou em inglês, *push*.

A remoção em uma pilha é *desempilhar* ou *pop*.



Após realizar as seguintes operações, de cima para baixo:

pop  
pop  
pop  
push  4  
push  5  
push 3  
pop  
push 4

A pilha irá conter os valores, do topo para baixo:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | 2,4,5,4 |
|  | B | 3,5,4,2  Você assinalou essa alternativa (B) |
|  | C | 2,4,3,5,4,5,4 |
|  | D | 4,5,4,2  **Aula 3 - Tema 3** |
|  | E | 4,3,5,4 |

Questão 8/10 - Programação III

Quanto a busca (pesquisa) em vetores são feitas as seguintes afirmativas:

 I. Na pesquisa sequencial, em um vetor ordenado, deve-se buscar o número até que seja encontrado e enquanto for maior que o número do vetor.

II. Na pesquisa binária, em um vetor não ordenado deve-se buscar o número até que seja encontrado ou até o fim do vetor

III.A pesquisa sequencial é mais eficiente que a pesquisa binária, visto que a pesquisa sequencial funciona tanto par vetores ordenados quanto para vetores desordenados.

Estão corretas as afirmativas:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | I somente.  **Aula 1 – tema 1** |
|  | B | II somente. |
|  | C | I e II somente. |
|  | D | I e III somente.  Você assinalou essa alternativa (D) |
|  | E | II e III somente. |

Questão 9/10 - Programação III

Observe a descrição abaixo:

* Comparações são feitas entre elementos de um vetor
* Cada elemento de uma posição i é comparado com o elemento da posição i + 1
* Se a ordenação procurada é encontrada é feita uma troca de posições entre os elementos

A descrição em questão se trata do algoritmo de ordenação:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Mergesort |
|  | B | Quicksort |
|  | C | Bubblesort  Você assinalou essa alternativa (C)  **Você acertou!**  **Aula 2 – tema 1** |
|  | D | Hashsort |
|  | E | Insertion sort |

Questão 10/10 - Programação III

Considere o algoritmo abaixo:

def algoritmo(dados):  
    tam = len(dados   
    for v in range(0, tam, 1):

        flag = 0  
        for i in range(0, tam - 1, 1):  
              if dados[i] < dados[i + 1]:

                 aux = dados[i]  
                 dados[i] = dados[i + 1]  
                 dados[i + 1] = aux  
                 flag = 1  
        if flag == 0:  
            return dados

Após análise do algoritmo acima, assinale a alternativa correta:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | O código é um  algoritmo de seleção e ordena em ordem crescente. |
|  | B | O código em questão é de um algoritmo de pesquisa e busca o menor número. |
|  | C | O código em questão  é de um algoritmo de pesquisa e busca o maior número. |
|  | D | O código em questão é de um algoritmo de ordenação e ordena em ordem crescente.  Você assinalou essa alternativa (D) |
|  | E | O código em questão é de um algoritmo de ordenação e ordena em ordem decrescente.  **Aula 2 – tema 1** |

Questão 1/10 - Programação III

Assuma uma lista com 10  dados numéricos e inteiros colocados na seguinte ordem:  
*[ 05 , 07 ,08 , 14 , 24 , 29 , 56, 77 , 78 , 88 ]*

Suponha que você deseja implementar um algoritmo de busca para localizar algum dado neste vetor já ordenado de maneira crescente. Você resolve testar a busca sequencial e a busca binária. (Adaptada)

Acerca destes algoritmos e analisando o vetor acima, assinale a alternativa CORRETA:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | No algoritmo de busca sequencial, o valor 24 seria localizado na 6ª tentativa, se fizermos uma varredura da esquerda para a direita.  Na 5ª tentativa. |
|  | B | No algoritmo de busca binária, o valor 24 seria localizado na 3ª tentativa.  Na 1ª tentativa. |
|  | C | No algoritmo de busca sequencial, o valor 77 seria localizado mais rapidamente que se comparado com a busca binária.  Binária se sairia mais rápida. |
|  | D | No algoritmo de busca sequencial, o valor 07 seria localizado mais rapidamente que se comparado com a busca binária.  Você assinalou essa alternativa (D)  **Você acertou!**  Levará menos iterações.  Aula1 tema1 |
|  | E | Em nenhum cenário de busca o algoritmo sequencial irá localizar o valor antes da busca binária.  É possível que sim, a sequencial ache antes. Dependerá o valor buscado e onde ele se localizado no vetor. |

Questão 2/10 - Programação III

Considere o algoritmo abaixo:

def algoritmo(dados):  
    tam = len(dados   
    for v in range(0, tam, 1):

        flag = 0  
        for i in range(0, tam - 1, 1):  
              if dados[i] < dados[i + 1]:

                 aux = dados[i]  
                 dados[i] = dados[i + 1]  
                 dados[i + 1] = aux  
                 flag = 1  
        if flag == 0:  
            return dados

Após análise do algoritmo acima, assinale a alternativa correta:

Nota: 0.0Você não pontuou essa questão

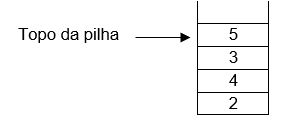
|  |  |  |
| --- | --- | --- |
|  | A | O código é um  algoritmo de seleção e ordena em ordem crescente.  Você assinalou essa alternativa (A) |
|  | B | O código em questão é de um algoritmo de pesquisa e busca o menor número. |
|  | C | O código em questão  é de um algoritmo de pesquisa e busca o maior número. |
|  | D | O código em questão é de um algoritmo de ordenação e ordena em ordem crescente. |
|  | E | O código em questão é de um algoritmo de ordenação e ordena em ordem decrescente.  **Aula 2 – tema 1** |

Questão 3/10 - Programação III

Considere a estrutura de dados do tipo *Pilha* abaixo.  Nessa pilha  existem quatro valores armazenados. O topo é indicado pelo ponteiro Topo da pilha.

A inserção em uma pilha é chamada de *empilhar*, ou em inglês, *push*.

A remoção em uma pilha é *desempilhar* ou *pop*.



Após realizar as seguintes operações, de cima para baixo:

pop  
pop  
pop  
push  4  
push  5  
push 3  
pop  
push 4

A pilha irá conter os valores, do topo para baixo:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | 2,4,5,4 |
|  | B | 3,5,4,2 |
|  | C | 2,4,3,5,4,5,4 |
|  | D | 4,5,4,2  Você assinalou essa alternativa (D)  **Você acertou!**  **Aula 3 - Tema 3** |
|  | E | 4,3,5,4 |

Questão 4/10 - Programação III

Observe o algoritmo abaixo:

def busca(x:list,alvo: int):

    n = len(x)  
    for i in range(0, n, 1):  
        if n[i] == alvo:  
            return i  
      #asserção: todos os elementos foram testados e alvo não foi encontrado.  
    return -1

Koffman, Elliot, B. e Paul A. T. Wolfgang. Objetos, Abstração, Estrutura de Dados e Projeto Usando C++. Disponível em: Minha Biblioteca, Grupo GEN, 2008.Modificado

Quanto ao algoritmo acima é correto afirmar:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | A complexidade assintótica é O(n).  Você assinalou essa alternativa (A)  **Você acertou!**  **Aula 1 -  tema 3** |
|  | B | A complexidade assintótica é O(i). |
|  | C | A complexidade assintótica é O(alvo). |
|  | D | A complexidade assintótica é O(n+1). |
|  | E | A complexidade assintótica é O(1). |

Questão 5/10 - Programação III

Quanto a busca (pesquisa) em vetores são feitas as seguintes afirmativas:

 I. Na pesquisa sequencial, em um vetor ordenado, deve-se buscar o número até que seja encontrado e enquanto for maior que o número do vetor.

II. Na pesquisa binária, em um vetor não ordenado deve-se buscar o número até que seja encontrado ou até o fim do vetor

III.A pesquisa sequencial é mais eficiente que a pesquisa binária, visto que a pesquisa sequencial funciona tanto par vetores ordenados quanto para vetores desordenados.

Estão corretas as afirmativas:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | I somente.  Você assinalou essa alternativa (A)  **Você acertou!**  **Aula 1 – tema 1** |
|  | B | II somente. |
|  | C | I e II somente. |
|  | D | I e III somente. |
|  | E | II e III somente. |

Questão 6/10 - Programação III

Quanto a análise de algoritmos assinale a alternativa correta:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Quando se fala algoritmo mais eficiente para resolver um problema, estamos nos referindo ao algoritmo de maior complexidade. |
|  | B | A complexidade de tempo analise a quantidade de instruções do código, não impactando diretamente em seu desempenho. |
|  | C | A quantidade de variáveis em um algoritmo e seus tamanhos impactam diretamente em seu desempenho.  Você assinalou essa alternativa (C)  **Você acertou!**  **Aula 1 – tema 2** |
|  | D | Quando um algoritmo  é  complexo em tempo,  ele é necessariamente complexo também em espaço. |
|  | E | A análise de algoritmos é altamente dependentemente do *hardware*. |

Questão 7/10 - Programação III

A técnica de divisão e conquista consistem de 3 passos básicos:

1.Divisão: Dividir o problema original, em subproblemas menores.   
2.Conquista: Resolver cada subproblema recursivamente.   
3.Combinação: Combinar as soluções encontradas, compondo uma solução para o problema original.

 É um exemplo de algoritmo de divisão e conquista:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Insertion sort |
|  | B | Mergesort  Você assinalou essa alternativa (B)  **Você acertou!**  Aula 2 tema 2 |
|  | C | Bubblesort |
|  | D | Selection sort |
|  | E | Searchsort |

Questão 8/10 - Programação III

Observe a descrição abaixo:

* Comparações são feitas entre elementos de um vetor
* Cada elemento de uma posição i é comparado com o elemento da posição i + 1
* Se a ordenação procurada é encontrada é feita uma troca de posições entre os elementos

A descrição em questão se trata do algoritmo de ordenação:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Mergesort |
|  | B | Quicksort |
|  | C | Bubblesort  Você assinalou essa alternativa (C)  **Você acertou!**  **Aula 2 – tema 1** |
|  | D | Hashsort |
|  | E | Insertion sort |

Questão 9/10 - Programação III

O algoritmo de ordenação pelo método da bolha (bubble sort ) foi utilizado para ordenar a sequência 43, 6, 32, 15, 21 de forma crescente.

Quantas trocas foram realizadas?

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | 2 |
|  | B | 3 |
|  | C | 4  Você assinalou essa alternativa (C) |
|  | D | 5 |
|  | E | 6  **Aula 2 – tema 1** |

Questão 10/10 - Programação III

Na aula 3 de nossa disciplina estudamos estruturas de dados que se comportam como uma FILA. (Adaptada)

Acerca de FILAS, assinale a alternativa CORRETA:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Uma fila onde o primeiro elemento é o 66, o segundo é o 33 e o terceiro é o 99. Inserir na fila significaria inserir um elemento que aponte para o valor 66. |
|  | B | Em uma fila, podemos ter a inserção dos dados no início desta fila. |
|  | C | Em uma fila, podemos ter a remoção dos dados final ou no meio desta fila. |
|  | D | Em uma fila trabalhamos com o conceito de: “o primeiro que entra é o primeiro que sai”.  Você assinalou essa alternativa (D)  **Você acertou!**  Aula 3 – tema 4 |
|  | E | Uma fila onde o primeiro elemento é o 66, o segundo é o 33 e o terceiro é o 99. Remover da fila significaria remover o elemento 99.  Removeria o 66 (remoção no início da fila). |

Questão 1/10 - Programação III

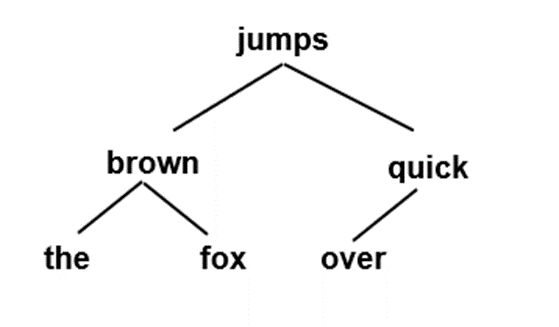
**Uma razão para estudarmos grafos é encontrar um caminho entre vértices. Quanto a vértices e caminhos, assinale a sentença correta.**

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | **Em um caminho simples, os vértices e arestas são distintos, exceto que o primeiro e o último vértices podem ser o mesmo.**  Você assinalou essa alternativa (A)  **Você acertou!**  **Aula 6 tema 1** |
|  | B | **Um ciclo é um caminho simples em que apenas o primeiro e o último vértices  estão conectados.** |
|  | C | **Um caminho é uma seqüência de arestas em que cada aresta adjacente é paralela ao seu predecessor.** |
|  | D | **Um nó vizinho de um vértice não pode estar conectado a outro vértice distinto.** |
|  | E | Em um grafo não orientado, um ciclo deve conter no mínimo quatro vértices. |

Questão 2/10 - Programação III

Considere a seguinte árvore binária.



Koffman, Elliot, B. e Paul A. T. Wolfgang. Objetos, Abstração, Estrutura de Dados e Projeto Usando C++. Disponível em: Minha Biblioteca, Grupo GEN, 2008.(Adaptado)

Qual é a ordem de visita seguindo a consulta em ordem?

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | jumps, brown,quick,the,fox,over |
|  | B | the, fox, brown, jumps, quick ,over |
|  | C | the, brown,fox,jumps,over,quick  Você assinalou essa alternativa (C)  **Você acertou!**  Aula 4 – tema 3 |
|  | D | jumps, brown,the,fox,quick,over |
|  | E | over,quick,jumps,fox,the,brown |

Questão 3/10 - Programação III

Dois matemáticos russos, G. M. Adel’son-Vel’skiî e E. M. Landis, publicaram em 1962 um artigo que descreve um algoritmo para manter o equilíbrio global de uma árvore de busca binária. Seu algoritmo controla a diferença de altura das subárvores. À medida que itens são adicionados à árvore (ou removidos dela), o fator de balanceamento\*\* (isto é, a diferença entre as alturas das subárvores) de cada subárvore do ponto de inserção até a raiz é mantido.

Koffman, Elliot, B. e Paul A. T. Wolfgang. Objetos, Abstração, Estrutura de Dados e Projeto Usando C++. Disponível em: Minha Biblioteca, Grupo GEN, 2008.

No caso de uma arvore AVL balanceada, o fator de balanceamento sempre será:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | menor ou igual a 2. |
|  | B | igual a 0 ou -1. |
|  | C | igual a -1, 0 ou 1.  Você assinalou essa alternativa (C)  **Você acertou!**  Aula 4 – tema 4 |
|  | D | maior que 1. |
|  | E | igual a 1. |

Questão 4/10 - Programação III

“... após o hash, duas chaves podem ser mapeadas para a mesma posição. Chamamos essa situação de colisão. Felizmente, existem técnicas eficazes para resolver o conflito criado por colisões.”

CORMEN, Thomas. Algoritmos - Teoria e Prática.Grupo GEN, 2012. E-book. ISBN 9788595158092. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/9788595158092/. PAG 186

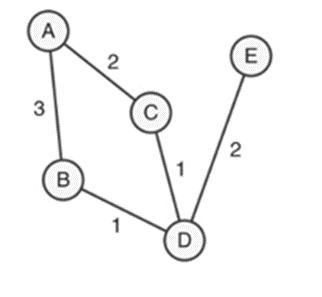
A maneira como é tratada as colisões depende muito do tipo de endereçamento.Acerca dos tipos de endereçamento, assinale a alternativa CORRETA:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | O endereçamento aberto é mais empregado quando a quantidade de palavras-chaves é bastante grande se comparado com o tamanho da tabela hash. |
|  | B | No endereçamento aberto a tabela hash é construída com um vetor, que armazenará todas as chaves que não colidirem. |
|  | C | No endereçamento aberto, quando uma colisão ocorre, ela precisa ser tratada com algum algoritmo, como o de tentativa linear e a quadrática.  Você assinalou essa alternativa (C)  **Você acertou!**  Aula 5 – TEMAS 3 e 4 |
|  | D | No endereçamento em cadeia não precisamos tratar colisões, pois cada nova chave pode ser anexada em uma lista encadeada que contém todas as chaves que colidiram. |
|  | E | As funções de hash aplicadas para endereçamento em cadeia são diferentes das aplicadas no endereçamento aberto. |

Questão 5/10 - Programação III

"Matematicamente, um grafo é um conjunto V de vértices e um conjunto E de arestas, de modo que cada aresta em E conecta dois dos vértices em V. O termo nó também é usado aqui como sinônimo de vértice. Vértices e arestas podem ser rotulados ou não rotulados. Quando as arestas são rotuladas com números, os números podem ser vistos como pesos, e o grafo é considerado um grafo ponderado."  
LAMBERT, Kenneth A. Fundamentos de Python: estruturas de dados. Ed. Cengage Learning Brasil, 2022. .pag 356

Observe a figura abaixo:  


Considerando o texto base  e a figura acima, são feitas as seguintes afirmativas:

I.Temos 4 vértices e o conjunto de vértices V é (A,B,C,D,E)

II.Temos 5 arestas e o conjunto de arestas E é (1,2,3)

III. O Grafo é  um grafo dirigido pois possui números nas arestas

Estão corretas as afirmativas:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | I apenas  Aula 6 – tema 1 |
|  | B | II apenas |
|  | C | I e II apenas  Você assinalou essa alternativa (C) |
|  | D | I e III apenas |
|  | E | II e III  apenas |

Questão 6/10 - Programação III

“Para representar grafos, você precisa de uma maneira conveniente de armazenar os vértices e as arestas que os conectam. “

LAMBERT, Kenneth A. Fundamentos de Python: estruturas de dados. Cengage Learning Brasil, 2022. E-book. ISBN 9786555584288. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/9786555584288/. Acesso em: 05 dez. 2022.pag 361

Na aula de grafos vimos duas representações de grafos comumente usadas. Assinale a alternativa que contém essas duas representações distintas:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | hash e lista ligada. |
|  | B | lista de adjacência e matriz  de adjacência.  Você assinalou essa alternativa (B)  **Você acertou!**  Aula 6 – tema 2 |
|  | C | matriz de adjacência e fila. |
|  | D | lista de adjacência e pilha. |
|  | E | lista dinâmica e lista de incidência. |

Questão 7/10 - Programação III

"Um grafo  G é uma estrutura composta de um conjunto finito de vértices V e um conjunto de arestas A V X V. Há duas formas básicas de representar um grafo G = (V, A): matriz de adjacências e listas de adjacências."  
Pereira, Silvio do Lago. Estruturas de dados em C : uma abordagem didática / Silvio do Lago Pereira. - São Paulo : Érica, 2016.Pag 185

Acerca do grafo e suas representações matemáticas, assinale a alternativa **INCORRETA.**

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Na representação por lista de adjacências, temos um conjunto de listas encadeadas, onde cada lista conterá todos os vizinhos de um único vértice; |
|  | B | Uma representação por matriz de incidências representa um grafo na forma de uma matriz, onde as linhas são os vértices e as colunas as arestas; |
|  | C | Uma representação por matriz de adjacências representa um grafo na forma de uma matriz, onde as linhas e as colunas são os vértices; |
|  | D | Uma representação por lista de adjacências representa um grafo na forma de um conjunto de listas encadeadas; |
|  | E | Na representação por lista de adjacências não podemos repetir um vértice em duas listas encadeadas distintas.  Você assinalou essa alternativa (E)  **Você acertou!**  Aula 6 - tema 1 Podemos repetir, pois cada lista conterá todos os vizinhos de cada vértice. |

Questão 8/10 - Programação III

A definição de uma boa função hash é fundamental para termos uma tabela hash com um bom desempenho.

Acerca de funções hash, são feitas as seguintes afirmativas:  
I. Uma função hash necessita inserir dados que minimizem o número de colisões, reduzindo também o tempo gasto resolvendo colisões e reavendo os dados.II.Uma função hash apresenta sempre a mesma fórmula bem definida, e independe do tamanho do conjunto de dados, e dos tipos de dados-chave utilizados.  
III. A função hash que utiliza o método da divisão só pode ser aplicado para palavras-chave do tipo numérica.

Estão corretas as afirmativas:

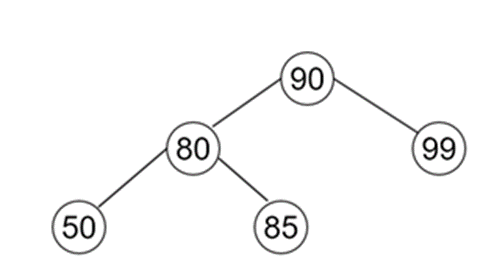
Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | I somente.  Você assinalou essa alternativa (A)  **Você acertou!**  Aula 5 – tema 2 |
|  | B | I e II somente. |
|  | C | I e III somente. |
|  | D | II e III somente |
|  | E | I, II e III. |

Questão 9/10 - Programação III

"A propriedade de auto balanceamento de uma árvore AVL é mantida por meio do fator de equilíbrio. Quando a diferença na altura das subárvores esquerda e direita atinge um valor maior do que 1 (ou menor do que - 1), a árvore precisa ser balanceada por meio de operações de rotação."  
Rodrigues, Thiago, N. et al. *Estrutura de Dados em Java*. Ed. Grupo A, 2021.pag 151

 Observe um exemplo de árvore AVL abaixo:

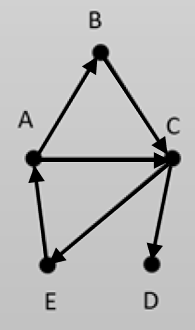


Suponha que você quer remover o nó folha de valor 99. Acerca do balanceamento e rotação desta árvore sem o 99. Assinale a alternativa CORRETA:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | A árvore ficará balanceada e não precisará de rotação nenhuma. |
|  | B | A árvore ficará com um desbalanceamento de valor 2 na raiz. |
|  | C | O nó filho de valor 80 está com balanceamento 0, resultando em uma rotação simples para a direta.  aula 4 - tema 4 Raiz -> Desbalanceada = -2. Filho da esquerda -> Balanceado = 0 Rotação simples para a direita |
|  | D | A árvore está com um desbalanceamento de valor -2 na raiz, resultando em uma rotação simples para a esquerda  Você assinalou essa alternativa (D) |
|  | E | O nó filho de valor 80 está com balanceamento 1, resultando em uma dupla com filho para a esquerda e pai para a direita. |

Questão 10/10 - Programação III

Observe o grafo:  
  


Considerando o grafo ilustrado acima, assinale a opção em que é apresentada a descrição em vértices (V) e arestas (A).

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | V = {A,B,C,D,E }  A = {(A, B), (A,C), (A, E), (C, E), (C, D),(E,A)} |
|  | B | V = {A,B,C,D,E }  A = {(A, B),(A,C), (A,E),(C, D), (C, E)} |
|  | C | V = {A,B,C,D,E }  A = {(E, A),(A,B) (B,C),  (C, E), (C, D)} |
|  | D | V = {A,B,C,D,E }  A = {(A, B), (B,C), (A, C), (C, E), (C, D),(E,A)}  Você assinalou essa alternativa (D)  **Você acertou!**  AULA 6 – TEMA 1 |
|  | E | V = {A,B,C,D,E }  A = {(C, E),(E,A),(A,B) (B,C),  (C, E), (C, D)} |

Questão 1/10 - Programação III

A definição de uma boa função hash é fundamental para termos uma tabela hash com um bom desempenho.

Acerca de funções hash, são feitas as seguintes afirmativas:  
I. Uma função hash necessita inserir dados que minimizem o número de colisões, reduzindo também o tempo gasto resolvendo colisões e reavendo os dados.II.Uma função hash apresenta sempre a mesma fórmula bem definida, e independe do tamanho do conjunto de dados, e dos tipos de dados-chave utilizados.  
III. A função hash que utiliza o método da divisão só pode ser aplicado para palavras-chave do tipo numérica.

Estão corretas as afirmativas:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | I somente.  Você assinalou essa alternativa (A)  **Você acertou!**  Aula 5 – tema 2 |
|  | B | I e II somente. |
|  | C | I e III somente. |
|  | D | II e III somente |
|  | E | I, II e III. |

Questão 2/10 - Programação III

Frequentemente, é útil determinar o caminho mais curto entre dois vértices em um grafo. Em aula foi visto o algoritmo de Dijkstra para  encontrar o caminho mais curto. Quanto ao algoritmos de Dijskstra visto em aula podemos afirmar:

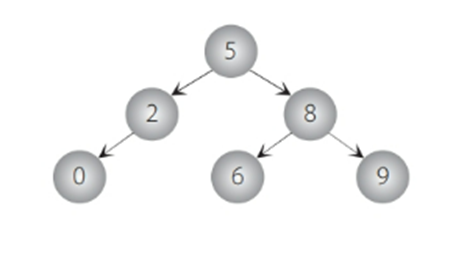
Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Algoritmo de Dijkstra considera apenas os pesos negativos para encontrar a maior rota. |
|  | B | Algoritmo de Dijskstra utiliza métrica aditiva, ou seja essa métrica encontra a maior rota considerando os pesos somados estre os caminhos. |
|  | C | Algoritmos de Dijkstra utiliza métrica preditiva, ou seja, essa métrica encontra a menor rota considerando o menor peso somado entre os caminhos. |
|  | D | Algoritmo de Dijkstra utiliza métrica aditiva, ou seja, essa métrica vai encontrar a menor rota considerando o menor peso somado entre os caminhos.  Você assinalou essa alternativa (D)  **Você acertou!**  **Aula 6 – tema 5** |
|  | E | Algoritmo de Dijkstra utiliza matriz de incidência para sua representação. |

Questão 3/10 - Programação III

Um percurso é uma forma sistemática de visitar e processar os nós de uma árvore.  
Um percurso em profundidade pode ser de três tipos básicos:  
Em-ordem: percorre a sub árvore esquerda, depois visita a raiz da árvore e, finalmente, percorre a sub árvore direita.  
Pré-ordem: visita a raiz da árvore, depois percorre a subárvore esquerda e, final- mente, percorre a subárvore direita.  
Pós-ordem: percorre a subárvore esquerda, depois percorre a subárvore direita e, finalmente, visita a raiz da árvore.

Pereira, Silvio do Lago. Estruturas de dados em C : uma abordagem didática / Silvio do Lago Pereira. - São Paulo : Érica, 2016. Pag 134 - modificado

Considere a seguinte arvore binária:  


Qual é a ordem de visita seguindo o percurso em pré ordem?

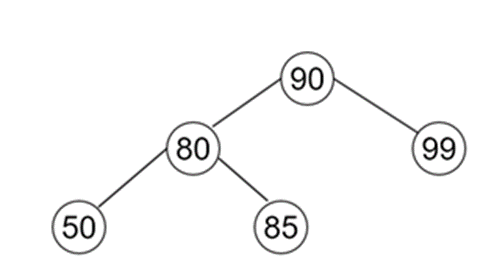
Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | 0,2,5,6,8,9  Você assinalou essa alternativa (A) |
|  | B | 0,2,6,9,8,5 |
|  | C | 0,2,6,8,9,5 |
|  | D | 5,2,0,8,6,9  Aula 4,  tema 3 |
|  | E | 5,2,8,0,6,9 |

Questão 4/10 - Programação III

"A propriedade de auto balanceamento de uma árvore AVL é mantida por meio do fator de equilíbrio. Quando a diferença na altura das subárvores esquerda e direita atinge um valor maior do que 1 (ou menor do que - 1), a árvore precisa ser balanceada por meio de operações de rotação."  
Rodrigues, Thiago, N. et al. *Estrutura de Dados em Java*. Ed. Grupo A, 2021.pag 151

 Observe um exemplo de árvore AVL abaixo:



Suponha que você quer remover o nó folha de valor 99. Acerca do balanceamento e rotação desta árvore sem o 99. Assinale a alternativa CORRETA:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | A árvore ficará balanceada e não precisará de rotação nenhuma. |
|  | B | A árvore ficará com um desbalanceamento de valor 2 na raiz. |
|  | C | O nó filho de valor 80 está com balanceamento 0, resultando em uma rotação simples para a direta.  aula 4 - tema 4 Raiz -> Desbalanceada = -2. Filho da esquerda -> Balanceado = 0 Rotação simples para a direita |
|  | D | A árvore está com um desbalanceamento de valor -2 na raiz, resultando em uma rotação simples para a esquerda |
|  | E | O nó filho de valor 80 está com balanceamento 1, resultando em uma dupla com filho para a esquerda e pai para a direita.  Você assinalou essa alternativa (E) |

Questão 5/10 - Programação III

Basicamente, a função  de hashing realiza um mapeamento dos registros de um arquivo por meio de um campo “chave”. A “chave” normalmente é determinada por um campo que possui um valor unívoco e, portanto, funciona como o identificador do arquivo, como, por exemplo, o RG de uma pessoa. Com esse mapeamento, um campo ou um conjunto de campos chaves e´ relacionado a um ou mais enderec¸os ou posições onde os registros estão  armazenados.

Lima, Diana M., D. e Luis E. F. Gonzalez. Matemática aplicada à informática (Tekne). Disponível em: Minha Biblioteca, Grupo A, 2015.pag57

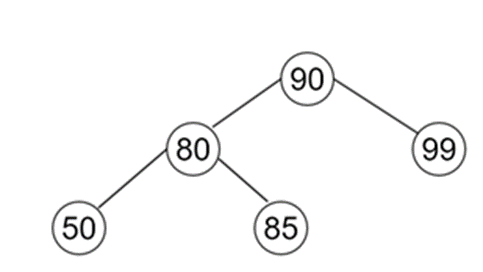
Acerca de hashs, vetores e tipos de endereçamento, assinale a alternativa **INCORRETA:**

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | O uso de tabela hash é capaz de transformar o tempo de busca de um dado em uma estrutura de dados do tipo vetor, em uma complexidade que independe do tamanho do conjunto de dados. |
|  | B | Podemos definir a posição de inserção de um dado no vetor utilizando uma função hash. Esta função será uma equação lógica e/ou matemática. |
|  | C | O endereçamento aberto em um vetor é aquele onde armazenamos um novo dado na primeira posição livre disponível no vetor.  Você assinalou essa alternativa (C) |
|  | D | O acesso a qualquer dado de um vetor com endereçamento direto é realizado com O(1), bem como o tempo de busca de uma informação neste vetor.  O tempo de busca com endereçamento direto é atrelado ao algoritmo de busca adotado. AULA 5 – TEMA 4. |
|  | E | Palavra-chave em uma tabela hash é aquele dado utilizado no cálculo de uma posição utilizando um algoritmo de hash. |

Questão 6/10 - Programação III

“Dois matemáticos russos, G. M. Adel’son-Vel’skiî e E. M. Landis, publicaram em 1962 um artigo que descreve um algoritmo para manter o equilíbrio global de uma árvore de busca binária. Seu algoritmo controla a diferença de altura das subárvores. À medida que itens são adicionados à árvore (ou removidos dela), o fator de balanceamento\*\* (isto é, a diferença entre as alturas das subárvores) de cada subárvore do ponto de inserção até a raiz é mantido. Se a um dado momento, o fator de balanceamento sair do intervalo -1 ... +1, a subárvore sofre uma rotação para retornar ao equilíbrio. As árvores que utilizam essa abordagem são conhecidas como árvores AVL, em homenagem aos seus inventores. Como antes, definimos a altura de uma árvore como o número de nós no maior caminho da raiz a um nó folha, inclusive a raiz.”

 KOFFMAN, Elliot B.; WOLFGANG, Paul A T. Objetos, Abstração, Estrutura de Dados e Projeto Usando C++. Grupo GEN, 2008. E-book. ISBN 978-85-216-2780-7. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/978-85-216-2780-7/. Acesso em: 07 dez. 2022. Capítulo 11.  
Observe um exemplo de árvore AVL abaixo:  


Considere  que o fator de balanceamento é calculado considerando  a árvore da direita – esquerda. Suponha que você quer remover o nó folha de valor 99. Acerca do balanceamento e rotação desta árvore sem o 99.

Assinale a alternativa CORRETA:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | A árvore ficará balanceada e não precisará de rotação nenhuma. |
|  | B | A árvore ficará com um desbalanceamento de valor 2 na raiz. |
|  | C | O nó filho de valor 80 está com balanceamento 0, resultando em uma rotação simples para a direta.  Raiz -> Desbalanceada = -2. Filho da esquerda -> Balanceado = 0 Rotação simples para a direita  Aula 4 – tema 4 |
|  | D | A árvore está com um desbalanceamento de valor -2 na raiz, resultando em uma rotação simples para a esquerda.  Você assinalou essa alternativa (D) |
|  | E | O nó filho de valor 80 está com balanceamento 1, resultando em uma dupla com filho para a esquerda e pai para a direita. |

Questão 7/10 - Programação III

**Uma razão para estudarmos grafos é encontrar um caminho entre vértices. Quanto a vértices e caminhos, assinale a sentença correta.**

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | **Em um caminho simples, os vértices e arestas são distintos, exceto que o primeiro e o último vértices podem ser o mesmo.**  Você assinalou essa alternativa (A)  **Você acertou!**  **Aula 6 tema 1** |
|  | B | **Um ciclo é um caminho simples em que apenas o primeiro e o último vértices  estão conectados.** |
|  | C | **Um caminho é uma seqüência de arestas em que cada aresta adjacente é paralela ao seu predecessor.** |
|  | D | **Um nó vizinho de um vértice não pode estar conectado a outro vértice distinto.** |
|  | E | Em um grafo não orientado, um ciclo deve conter no mínimo quatro vértices. |

Questão 8/10 - Programação III

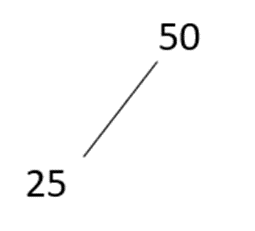
Existem duas ordens comuns nas quais os vértices podem ser visitados durante o percurso em um grafo. O primeiro, chamado de percurso em profundidade, o segundo tipo de percurso, chamado de percurso em largura.

O percurso em largura em grafos utiliza qual estrutura de dados?

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | Pilha |
|  | B | Fila  Você assinalou essa alternativa (B)  **Você acertou!**  Aula 6 – tema 4 |
|  | C | Hash |
|  | D | Dicionário |
|  | E | Árvore |

Questão 9/10 - Programação III

Observe a seguinte arvore AVL:  


Ao se inserir o valor 40 nessa árvore, as seguintes afirmativas são feitas:

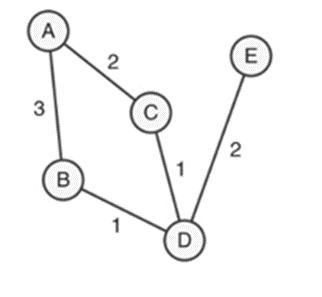
I. O fator de balanceamento da raiz 50 passa a ser -2 e a árvore fica desbalanceada.  
II. A árvore fica desbalanceada e uma rotação simples à direita é suficiente para balancear a árvore.  
III. A arvore fica desbalanceada e uma rotação dupla a direita (rotação esquerda-direita) é necessária.

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | I somente. |
|  | B | I e II somente.  Você assinalou essa alternativa (B) |
|  | C | I e III somente.  Aula 4 – tema 5 |
|  | D | II e III somente. |
|  | E | I, II e III. |

Questão 10/10 - Programação III

"Matematicamente, um grafo é um conjunto V de vértices e um conjunto E de arestas, de modo que cada aresta em E conecta dois dos vértices em V. O termo nó também é usado aqui como sinônimo de vértice. Vértices e arestas podem ser rotulados ou não rotulados. Quando as arestas são rotuladas com números, os números podem ser vistos como pesos, e o grafo é considerado um grafo ponderado."  
LAMBERT, Kenneth A. Fundamentos de Python: estruturas de dados. Ed. Cengage Learning Brasil, 2022. .pag 356

Observe a figura abaixo:  


Considerando o texto base  e a figura acima, são feitas as seguintes afirmativas:

I.Temos 4 vértices e o conjunto de vértices V é (A,B,C,D,E)

II.Temos 5 arestas e o conjunto de arestas E é (1,2,3)

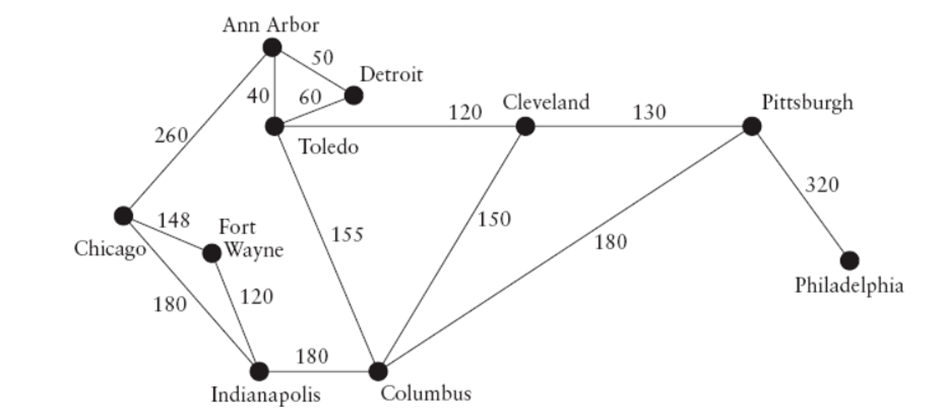
III. O Grafo é  um grafo dirigido pois possui números nas arestas

Estão corretas as afirmativas:

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | I apenas  Aula 6 – tema 1 |
|  | B | II apenas  Você assinalou essa alternativa (B) |
|  | C | I e II apenas |
|  | D | I e III apenas |
|  | E | II e III  apenas |

Questão 1/10 - Programação III

Observe a figura abaixo:  
  


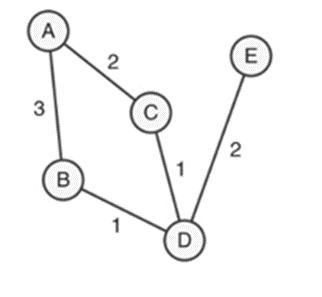
**Podemos afirmar que a figura pode ser representada por uma estrutura de dados. Qual a estrutura de dados que melhor representa a figura acima?**

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | **Árvore AVL** |
|  | B | **Grafo**  Você assinalou essa alternativa (B)  **Você acertou!**  Aula 6 - tema 1 |
|  | C | **Heap** |
|  | D | **Hash** |
|  | E | Fila |

Questão 2/10 - Programação III

"Matematicamente, um grafo é um conjunto V de vértices e um conjunto E de arestas, de modo que cada aresta em E conecta dois dos vértices em V. O termo nó também é usado aqui como sinônimo de vértice. Vértices e arestas podem ser rotulados ou não rotulados. Quando as arestas são rotuladas com números, os números podem ser vistos como pesos, e o grafo é considerado um grafo ponderado."  
LAMBERT, Kenneth A. Fundamentos de Python: estruturas de dados. Ed. Cengage Learning Brasil, 2022. .pag 356

Observe a figura abaixo:  


Considerando o texto base  e a figura acima, são feitas as seguintes afirmativas:

I.Temos 4 vértices e o conjunto de vértices V é (A,B,C,D,E)

II.Temos 5 arestas e o conjunto de arestas E é (1,2,3)

III. O Grafo é  um grafo dirigido pois possui números nas arestas

Estão corretas as afirmativas:

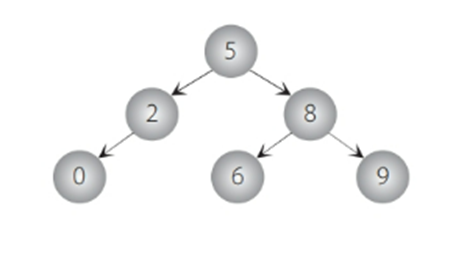
Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | I apenas  Aula 6 – tema 1 |
|  | B | II apenas |
|  | C | I e II apenas |
|  | D | I e III apenas |
|  | E | II e III  apenas  Você assinalou essa alternativa (E) |

Questão 3/10 - Programação III

Um percurso é uma forma sistemática de visitar e processar os nós de uma árvore.  
Um percurso em profundidade pode ser de três tipos básicos:  
Em-ordem: percorre a sub árvore esquerda, depois visita a raiz da árvore e, finalmente, percorre a sub árvore direita.  
Pré-ordem: visita a raiz da árvore, depois percorre a subárvore esquerda e, final- mente, percorre a subárvore direita.  
Pós-ordem: percorre a subárvore esquerda, depois percorre a subárvore direita e, finalmente, visita a raiz da árvore.

Pereira, Silvio do Lago. Estruturas de dados em C : uma abordagem didática / Silvio do Lago Pereira. - São Paulo : Érica, 2016. Pag 134 - modificado

Considere a seguinte arvore binária:  


Qual é a ordem de visita seguindo o percurso em pré ordem?

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | 0,2,5,6,8,9 |
|  | B | 0,2,6,9,8,5 |
|  | C | 0,2,6,8,9,5 |
|  | D | 5,2,0,8,6,9  Aula 4,  tema 3 |
|  | E | 5,2,8,0,6,9  Você assinalou essa alternativa (E) |

Questão 4/10 - Programação III

Basicamente, a função  de hashing realiza um mapeamento dos registros de um arquivo por meio de um campo “chave”. A “chave” normalmente é determinada por um campo que possui um valor unívoco e, portanto, funciona como o identificador do arquivo, como, por exemplo, o RG de uma pessoa. Com esse mapeamento, um campo ou um conjunto de campos chaves e´ relacionado a um ou mais enderec¸os ou posições onde os registros estão  armazenados.

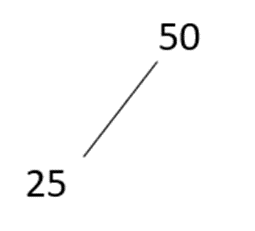
Lima, Diana M., D. e Luis E. F. Gonzalez. Matemática aplicada à informática (Tekne). Disponível em: Minha Biblioteca, Grupo A, 2015.pag57

Acerca de hashs, vetores e tipos de endereçamento, assinale a alternativa **INCORRETA:**

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | O uso de tabela hash é capaz de transformar o tempo de busca de um dado em uma estrutura de dados do tipo vetor, em uma complexidade que independe do tamanho do conjunto de dados.  Você assinalou essa alternativa (A) |
|  | B | Podemos definir a posição de inserção de um dado no vetor utilizando uma função hash. Esta função será uma equação lógica e/ou matemática. |
|  | C | O endereçamento aberto em um vetor é aquele onde armazenamos um novo dado na primeira posição livre disponível no vetor. |
|  | D | O acesso a qualquer dado de um vetor com endereçamento direto é realizado com O(1), bem como o tempo de busca de uma informação neste vetor.  O tempo de busca com endereçamento direto é atrelado ao algoritmo de busca adotado. AULA 5 – TEMA 4. |
|  | E | Palavra-chave em uma tabela hash é aquele dado utilizado no cálculo de uma posição utilizando um algoritmo de hash. |

Questão 5/10 - Programação III

Observe a seguinte arvore AVL:  


Ao se inserir o valor 40 nessa árvore, as seguintes afirmativas são feitas:

I. O fator de balanceamento da raiz 50 passa a ser -2 e a árvore fica desbalanceada.  
II. A árvore fica desbalanceada e uma rotação simples à direita é suficiente para balancear a árvore.  
III. A arvore fica desbalanceada e uma rotação dupla a direita (rotação esquerda-direita) é necessária.

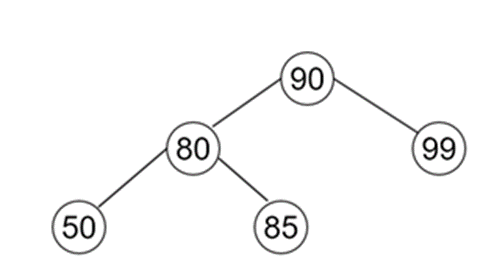
Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | I somente. |
|  | B | I e II somente. |
|  | C | I e III somente.  Você assinalou essa alternativa (C)  **Você acertou!**  Aula 4 – tema 5 |
|  | D | II e III somente. |
|  | E | I, II e III. |

Questão 6/10 - Programação III

"A propriedade de auto balanceamento de uma árvore AVL é mantida por meio do fator de equilíbrio. Quando a diferença na altura das subárvores esquerda e direita atinge um valor maior do que 1 (ou menor do que - 1), a árvore precisa ser balanceada por meio de operações de rotação."  
Rodrigues, Thiago, N. et al. *Estrutura de Dados em Java*. Ed. Grupo A, 2021.pag 151

 Observe um exemplo de árvore AVL abaixo:



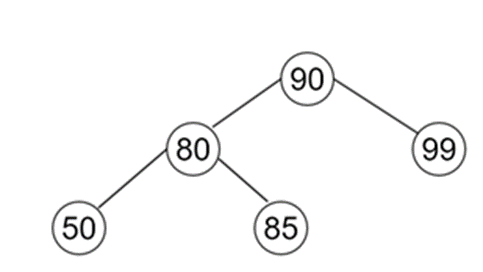
Suponha que você quer remover o nó folha de valor 99. Acerca do balanceamento e rotação desta árvore sem o 99. Assinale a alternativa CORRETA:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | A árvore ficará balanceada e não precisará de rotação nenhuma. |
|  | B | A árvore ficará com um desbalanceamento de valor 2 na raiz. |
|  | C | O nó filho de valor 80 está com balanceamento 0, resultando em uma rotação simples para a direta.  Você assinalou essa alternativa (C)  **Você acertou!**  aula 4 - tema 4 Raiz -> Desbalanceada = -2. Filho da esquerda -> Balanceado = 0 Rotação simples para a direita |
|  | D | A árvore está com um desbalanceamento de valor -2 na raiz, resultando em uma rotação simples para a esquerda |
|  | E | O nó filho de valor 80 está com balanceamento 1, resultando em uma dupla com filho para a esquerda e pai para a direita. |

Questão 7/10 - Programação III

“Dois matemáticos russos, G. M. Adel’son-Vel’skiî e E. M. Landis, publicaram em 1962 um artigo que descreve um algoritmo para manter o equilíbrio global de uma árvore de busca binária. Seu algoritmo controla a diferença de altura das subárvores. À medida que itens são adicionados à árvore (ou removidos dela), o fator de balanceamento\*\* (isto é, a diferença entre as alturas das subárvores) de cada subárvore do ponto de inserção até a raiz é mantido. Se a um dado momento, o fator de balanceamento sair do intervalo -1 ... +1, a subárvore sofre uma rotação para retornar ao equilíbrio. As árvores que utilizam essa abordagem são conhecidas como árvores AVL, em homenagem aos seus inventores. Como antes, definimos a altura de uma árvore como o número de nós no maior caminho da raiz a um nó folha, inclusive a raiz.”

 KOFFMAN, Elliot B.; WOLFGANG, Paul A T. Objetos, Abstração, Estrutura de Dados e Projeto Usando C++. Grupo GEN, 2008. E-book. ISBN 978-85-216-2780-7. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/978-85-216-2780-7/. Acesso em: 07 dez. 2022. Capítulo 11.  
Observe um exemplo de árvore AVL abaixo:  


Considere  que o fator de balanceamento é calculado considerando  a árvore da direita – esquerda. Suponha que você quer remover o nó folha de valor 99. Acerca do balanceamento e rotação desta árvore sem o 99.

Assinale a alternativa CORRETA:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | A árvore ficará balanceada e não precisará de rotação nenhuma. |
|  | B | A árvore ficará com um desbalanceamento de valor 2 na raiz. |
|  | C | O nó filho de valor 80 está com balanceamento 0, resultando em uma rotação simples para a direta.  Você assinalou essa alternativa (C)  **Você acertou!**  Raiz -> Desbalanceada = -2. Filho da esquerda -> Balanceado = 0 Rotação simples para a direita  Aula 4 – tema 4 |
|  | D | A árvore está com um desbalanceamento de valor -2 na raiz, resultando em uma rotação simples para a esquerda. |
|  | E | O nó filho de valor 80 está com balanceamento 1, resultando em uma dupla com filho para a esquerda e pai para a direita. |

Questão 8/10 - Programação III

“Para representar grafos, você precisa de uma maneira conveniente de armazenar os vértices e as arestas que os conectam. “

LAMBERT, Kenneth A. Fundamentos de Python: estruturas de dados. Cengage Learning Brasil, 2022. E-book. ISBN 9786555584288. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/9786555584288/. Acesso em: 05 dez. 2022.pag 361

Na aula de grafos vimos duas representações de grafos comumente usadas. Assinale a alternativa que contém essas duas representações distintas:

Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | hash e lista ligada. |
|  | B | lista de adjacência e matriz  de adjacência.  Você assinalou essa alternativa (B)  **Você acertou!**  Aula 6 – tema 2 |
|  | C | matriz de adjacência e fila. |
|  | D | lista de adjacência e pilha. |
|  | E | lista dinâmica e lista de incidência. |

Questão 9/10 - Programação III

"Uma árvore binária de busca tem a seguinte propriedade: para cada nó n da árvore, todos os valores armazenados em sua subárvore à esquerda (a árvore cuja raiz é o filho à es-querda) são menores que o valor v armazenado em n, e todos os valores armazenados na subárvore à direita são maiores ou igual a v."

DROZDEK, Adam. Estrutura de Dados e Algoritmos em C++ – Tradução da 4ª edição norte-americana. Cengage Learning Brasil, 2018. E-book. ISBN 9788522126651. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/9788522126651/. Acesso em: 07 dez. 2022.PAG189

Observe o código abaixo:  
  
1 class BST:  
2    def \_\_init\_\_(self,dado=None):  
3        self.dado = dado  
4        self.esquerda = None  
5        self.direita = None  
6  
7    def inserir(self, dado):  
8        if(self.dado ==None):  
9            self.dado = dado  
10        else:  
11            if(dado <self.dado):  
12                if(self.esquerda):  
13                    self.esquerda.inserir(dado)  
14                else:  
15                    self.esquerda = BST(dado)  
16            else:  
17                if(self.direita):  
18                    self.direita.inserir(dado)  
19                else:  
20                    self.direita = BST(dado)  
  
Considerando o texto, o conteúdo visto em aula e o código acima, assinale a alternativa **INCORRETA:**

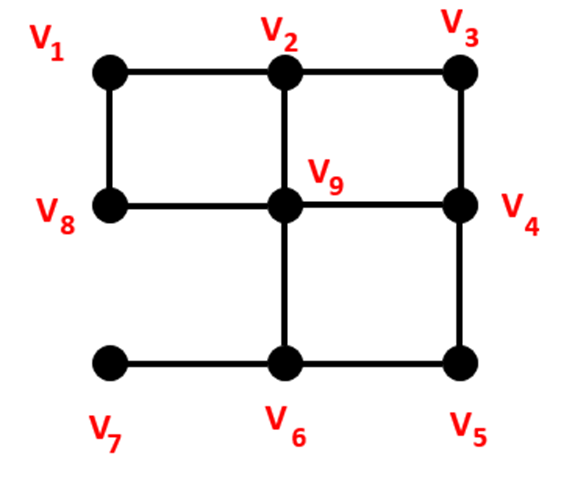
Nota: 10.0

|  |  |  |
| --- | --- | --- |
|  | A | O trecho de código que representa a inserção do nó raiz está nas linhas 8 e 9. |
|  | B | O trecho de código que  armazena os nós maiores que a raiz é representado pelas linhas 16 a 20. |
|  | C | O trecho de código que armazena os nós menores que a raiz é representado pelas linhas de 8 a 13.  Você assinalou essa alternativa (C)  **Você acertou!**  **O trecho de código que armazena os nós menores que a raiz é representado pelas linhas de 11 a 15**  **Aula 4 – tema 2** |
|  | D | A função init é um construtor da classe BST, inicializando as variáveis esquerda e direita com o valor None. |
|  | E | A função inserir é uma função recursiva, sendo chamada nas linhas 13 e 18. |

Questão 10/10 - Programação III

“Visto de forma abstrata, um grafo G e´ simplesmente um conjunto V de ve´rtices e uma colec¸a~o E de pares de ve´rtices de V, chamados de arestas. Assim, um grafo e´ uma forma de representar conexo~es ou relac¸o~es entre pares de objetos de algum conjunto V.”

GOODRICH, Michael T.; TAMASSIA, Roberto. **Estruturas de Dados e Algoritmos em Java**.Grupo A, 2013. pag 613

Abaixo temos uma imagem de um grafo.  


Acerca do grafo acima, considerando o texto base e o conteúdo visto em aula, assinale a alternativa CORRETA.

Nota: 0.0Você não pontuou essa questão

|  |  |  |
| --- | --- | --- |
|  | A | O grafo contém arestas múltiplas, pois temos mais de um caminho para sair de V1 e chegar em V9, por exemplo |
|  | B | O grau do vértice V9 é 3.  Você assinalou essa alternativa (B) |
|  | C | Todos os vértices deste grafo têm o mesmo grau. |
|  | D | Este grafo é do tipo completo. |
|  | E | O grau do vértice V4 é 3.  Aula 6 – Tema 1 |