



Avaliando Aprendizado

Teste seu conhecimento acumulado

Disc.: **ESTRUTURA DE DADOS**

Acertos: **2,0** de 2,0

23/11/2023



1^a

Questão /

Acerto: **0,2 / 0,2**

(IF-RS/2016 - Adaptada) Estrutura de dados é um conceito fundamental na ciência da computação e na programação. Ela se refere a maneira como os dados são organizados e armazenados em um programa ou sistema, a fim de que possam ser acessados, manipulados e utilizados de forma eficiente. Referente a alocação dinâmica de memória em C, é correto afirmar:

- ☒ ☒ As funções malloc e free e o operador sizeof, são essenciais para a alocação dinâmica de memória.
- ☐ A função free é geralmente usada com o operador sizeof.
- ☐ A função clear é usada para limpar o conteúdo de um ponteiro.
- ☐ As funções calloc e realloc são usadas para liberar arrays.
- ☐ A função malloc usa o número de blocos de memória que serão alocados na memória.

Respondido em 23/11/2023 10:00:19

Explicação:

A alocação dinâmica de memória em C é uma técnica importante que permite que o programador aloque memória em tempo de execução. As funções malloc e free são usadas para alocar e liberar a memória, respectivamente. Já o operador sizeof é usado para determinar o tamanho em bytes de um tipo de dado ou de uma variável.



2^a

Questão /

Acerto: **0,2 / 0,2**

Há duas maneiras de se passar argumentos ou parâmetros para funções: por valor e por referência. Sobre passagem de parâmetros, analise as seguintes afirmativas:

- I. Na passagem por referência, o que é passado como argumento no parâmetro formal é o endereço da variável.
- II. Na passagem por valor, o valor é copiado do argumento para o parâmetro formal da função.
- III. Por exemplo, quando duas variáveis inteiras i1 e i2 são passadas por valor à função troca() chamada pelo programa principal, elas também são alteradas no programa principal.
- IV. Na passagem por referência, dentro da função, o argumento real utilizado na chamada é acessado através do seu endereço, sendo assim alterado.
- V. Na passagem por valor, quaisquer alterações feitas nestes parâmetros dentro da função não irão afetar as variáveis usadas como argumentos para chamá-la.

Está CORRETO o que se afirma em:

- ☐ I e III
- ☐ II e IV, apenas
- ☐ I, III e V, apenas
- ☐ V, apenas
- ☒ I, II, IV e V, apenas

Respondido em 23/11/2023 10:20:31

Explicação:

Resposta correta: I, II, IV e V, apenas



3ª

Questão /

Acerto: 0,2 / 0,2

Várias estruturas de dados podem ser utilizadas para armazenar dados de um aplicação. Em relação ao assunto, assinale a alternativa correta.

- ☐ A estrutura de dados do tipo pilha, sempre retira os elementos que foram inseridos primeiro na estrutura.
- ☒ A estrutura de dados do tipo fila, utiliza a ideia do primeiro a ser inserido, será o primeiro a ser retirado.
- ☐ A estrutura de dados do tipo lista, utiliza a ideia do primeiro a chegar, primeiro a ser servido para inserir elementos.
- ☐ A estrutura de dados do tipo fila, sempre retira os elementos que entraram por último na fila.
- ☐ A estrutura de dados do tipo pilha, para retirar o elemento do topo da pilha, é necessário retirar o elemento da base da pilha.

Respondido em 23/11/2023 10:03:51

Explicação:

A resposta correta é: A estrutura de dados do tipo fila, utiliza a ideia do primeiro a ser inserido, será o primeiro a ser retirado.



4ª

Questão /

Acerto: 0,2 / 0,2

(CESGRANRIO/2014) Considere utilizar o algoritmo Bubble Sort para ordenar, em ordem crescente, a sequência de números

17, 43, 37, 31, 8, 77, 52, 25.

Se a sequência original for a iteração zero, qual será a sequência de números da segunda iteração?

- ☐ 17, 31, 37, 43, 8, 77, 52, 55
- ☐ 17, 31, 8, 25, 37, 43, 77, 52
- ☐ 8, 17, 43, 37, 31, 77, 52, 25
- ☒ 17, 31, 8, 37, 43, 25, 52, 77
- ☐ 17, 25, 37, 31, 8, 43, 52, 77

Respondido em 23/11/2023 09:55:19

Explicação:

Para resolver essa questão, vamos simular a execução do algoritmo Bubble Sort na sequência dada e acompanhar as trocas feitas na segunda iteração.

Sequência original: 17, 43, 37, 31, 8, 77, 52, 25

Primeira iteração:

17, 37, 31, 8, 43, 52, 25, 77 (comparação e troca: 43 e 37)

17, 31, 8, 37, 43, 25, 52, 77 (comparação e troca: 37 e 31)

17, 31, 8, 37, 25, 43, 52, 77 (comparação e troca: 43 e 25)

17, 31, 8, 37, 25, 43, 52, 77 (sem trocas)

Segunda iteração (a partir da sequência obtida na primeira iteração):

17, 31, 8, 37, 25, 43, 52, 77 (comparação e troca: 31 e 8)

17, 8, 31, 37, 25, 43, 52, 77 (comparação e troca: 37 e 8)

17, 8, 31, 25, 37, 43, 52, 77 (sem trocas)

17, 8, 31, 25, 37, 43, 52, 77 (sem trocas)

A sequência obtida na segunda iteração é: 17, 8, 31, 25, 37, 43, 52, 77.

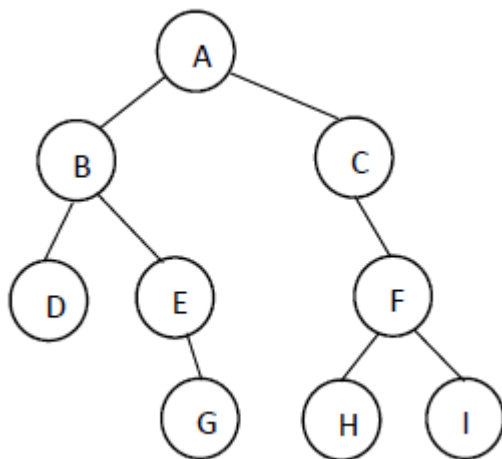


Questão /

Acerto: 0,2 / 0,2

Ano: 2020 Banca: Instituto AOCP Órgão: Prefeitura de Nova Hamburgo - RS Prova: Analista de Desenvolvimento - Analista de Sistemas

Analise a seguinte árvore binária e assinale a alternativa correta.



- ☒ TA é a subárvore enraizada em "A", portanto toda a árvore.
- ☐ Com exceção do nó "A", que é raiz, os demais nós são conhecidos como folhas.
- ☐ "A" é filho de todos.
- ☐ "B" tem grau de saída 3 e grau 2.
- ☐ "B" e "C" são caules da árvore.

Explicação:

Resposta correta: TA é a subárvore enraizada em "A", portanto toda a árvore.



6ª

Questão /

Acerto: 0,2 / 0,2

(COTEC/2013 - Adaptada) A linguagem C permite alocar (reservar) dinamicamente (em tempo de execução) blocos de memórias utilizando ponteiros. A esse processo dá-se o nome de alocação dinâmica, que faz uso das funções malloc, calloc, realloc e free, disponíveis na biblioteca stdlib.h. Para liberar um bloco de memória previamente alocado, por meio de um único parâmetro de entrada, faz-se uso de qual função?

- ☒ Free.
- ☐ Calloc.
- ☐ Malloc.
- ☐ Hiploc.
- ☐ Realloc.

Respondido em 23/11/2023 10:24:07

Explicação:

A função "free" é responsável por liberar a memória previamente alocada pelo programador com as funções "malloc", "calloc" ou "realloc". Ela recebe como parâmetro um ponteiro para o primeiro byte da memória alocada e libera toda a área de memória correspondente.



7ª

Questão /

Acerto: 0,2 / 0,2

Em relação ao uso e conceitos de procedimentos e funções em lógica de programação, analise as seguintes afirmativas:

- I. Procedimentos e funções são blocos de instruções para realizar tarefas específicas e são considerados sub-rotinas.
- II. Em um procedimento, a passagem de parâmetros é obrigatória.
- III. Em uma função, a passagem de parâmetros e o retorno de um valor são obrigatórios.

Está CORRETO o que se afirma em:

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

Respondido em 23/11/2023 10:25:05

Explicação:

Resposta correta: I, apenas.



8ª

Questão /

Acerto: 0,2 / 0,2

Uma pilha segue a regra: "o ultimo a chegar é o primeiro a sair". Já as filas obedecem à regra: o primeiro a chegar é o primeiro a sair. Com base nesses argumentos,

Uma pilha P e uma fila F originalmente com n elementos cada ($n > 5$), onde suas operações são:

empilha(P, elemento): insere elemento na pilha P;

desempilha(P): remove da pilha P e retorna o elemento removido;

enfileira(F, elemento): insere elemento na fila F;

desenfileira(F): remove da fila F e retorna o elemento removido;

para $i = 1$ até n, faça

empilha(P, desempilha(P))

enfileira(F, desenfileira(F))

fim-para

Ao final da execução do pseudocódigo, os estados finais de P e F serão respectivamente:

- ☐ Ambas as estruturas estarão vazias.
- ☒ elementos em ordem original e elementos em ordem original.
- ☐ elementos em ordem inversa e elementos em ordem inversa.
- ☐ elementos em ordem inversa e elementos em ordem original.
- ☐ elementos em ordem original e elementos em ordem inversa.

Respondido em 23/11/2023 10:26:53

Explicação:

A resposta correta é: elementos em ordem original e elementos em ordem original.



Questão /

Acerto: 0,2 / 0,2

Sobre o método da bolha é correto afirmar que:

- ☐ O algoritmo executa sempre no mesmo tempo para instâncias de mesmo tamanho n .
- ☐ O tempo de execução pode ser linear em relação ao tamanho da entrada se a instância apresentada estiver ordenada em ordem reversa a desejada.
- ☐ O tempo de execução é definido pela complexidade computacional sempre, independentemente da instância apresentada.
- ☒ O tempo de execução pode ser linear em relação ao tamanho da entrada se a instância apresentada já estiver ordenada.
- ☐ A complexidade computacional deste algoritmo é $O(n \log n)$.


Respondido em 23/11/2023 09:53:47

Explicação:

A resposta correta é: O tempo de execução pode ser linear em relação ao tamanho da entrada se a instância apresentada já estiver ordenada.



As árvores binárias de busca são especializações das árvores binárias que permitem uma melhor organização dos algoritmos de busca. Sobre a inserção de uma nova chave em uma árvore binária de busca é correto afirmar que:

- ☒  Todas as chaves são inseridas em folhas, a posição da folha é determinada pela busca.
- ☐ O algoritmo de inserção em árvores binárias de busca é estático, isto é, é necessário recalcular toda árvore para inserir uma nova chave.
- ☐ Para determinar a posição da nova chave é necessário calcular o percurso em ordem simétrica da árvore obtida. Com este percurso, verifica-se se a sequência está ordenada em ordem crescente. Caso esteja, a posição da nova chave está correta.
- ☐ Toda nova chave é inserida obrigatoriamente na raiz.
- ☐ A complexidade da inserção é sempre $O(n)$, independentemente da altura da árvore.

Respondido em 23/11/2023 10:27:56

Explicação:

Todas as chaves serão inseridas em folhas, considerando que na árvore binária de busca o objetivo é buscar pela posição correta de inserção dos dados e inserir (a esquerda ou a direita) o nó considerado, além de que em árvores binárias de busca não existem regras que mantenham o balanceamento da árvore.