

Lista 1 - Estrutura de Dados

Matéria: Revisão de C, Árvores binárias, ABB, AVL e Grafos

Revisão de C

Q1 - Escreva um programa que, infinitamente, leia um número inteiro, que representa o tamanho, seguida de uma sequência de números inteiros, e determine o comprimento máximo de um segmento crescente destes números. Exemplos: Na sequência 5, 10, 3, 2, 4, 7, 9, 8, 5 o comprimento do segmento crescente máximo é 4. Na sequência 10, 8, 7, 5, 2 o comprimento de um segmento crescente máximo é 1. Seu programa para quando n for menor ou igual a zero.

Q2 - Implemente um programa que, infinitamente, receba, como parâmetro de entrada, um número n e retorne a representação binária de n. Por exemplo, se n é igual a 12, a resposta deste programa deve ser "1100". Seu programa para quando n for menor que zero.

Q3- Seja a seguinte sequência de instruções em um programa C:

```
int *pti;  
int veti[]={10,7,2,6,3};  
pti = veti;
```

Qual afirmativa é **falsa**?

- a. *pti é igual a 10
- b. *(pti+2) é igual a 2
- c. pti[4] é igual a 3
- d. pti[1] é igual a 10
- e. *(veti+3) é igual a 6

Q3 - O que significa o operador asterisco em cada um dos seguintes casos:

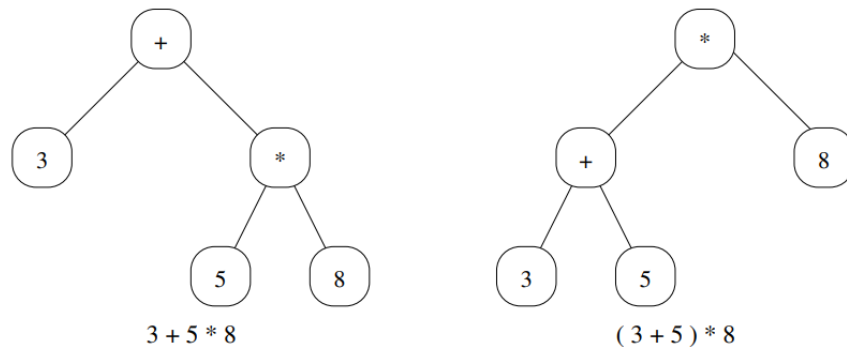
- a) int *p;
- b) cout << *p;
- c) *p = x*5;
- d) cout << *(p+1);

Árvores Binárias

Q4 - Quantos antecedentes tem um nó no nível n em uma árvore binária?

Q5 - Uma árvore estritamente binária com n nós folhas contém quantos nós?

Q6 - Considere árvores binárias que representam expressões aritméticas (composta por operandos compostos por um único algarismo, operações de +, -, * e / e parênteses) como as apresentadas abaixo.



Q7 - Escreva um algoritmo que receba uma expressão matemática (composta por operandos compostos por um único algarismo, operações de +, -, * e / e parênteses) representada por um string e retorne uma árvore binária representando esta expressão.

Árvores Binárias de Busca

Para as próximas questões, assuma que esta é a struct do nó da árvore.

```
typedef struct reg {
    int     chave;
    int     conteudo;
    struct reg *esq, *dir;
} noh;
```

Q8 - Suponha que $x \rightarrow \text{esq} \rightarrow \text{chave} \leq x \rightarrow \text{chave}$ para cada nó x dotado de filho esquerdo e $x \rightarrow \text{chave} \leq x \rightarrow \text{dir} \rightarrow \text{chave}$ para cada nó x dotado de filho direito. Essa árvore é de busca?

Q9 - Escreva uma função que decida se uma dada árvore binária é ou não é de busca.

Q10 - Suponha que as chaves 50 30 70 20 40 60 80 15 25 35 45 36 são inseridas, nesta ordem, numa árvore de busca inicialmente vazia. Desenhe a árvore que resulta. Em seguida remova o nó que contém 30.

Q11 - Considere árvores binárias de busca cujos nós têm a estrutura indicada abaixo. Escreva uma função que receba a raiz de uma tal árvore e o endereço de um nó x e devolva o endereço do pai de x .

```
typedef struct reg {
    int     chave;
```

```

int    conteudo;
struct reg *esq, *dir;
} noh;

```

Se x não pertence à árvore, a função deve devolver NULL. O consumo de tempo de sua função deve ser limitado pela profundidade de x .

Árvores AVL

Q12- Dada uma AVL cuja raiz é um nó folha com a chave 50:

- (a) insira os elementos {1, 64, 12, 18, 66, 38, 95, 58, 59, 70, 68, 39, 62, 7, 60, 43, 16, 67, 34, 35} nesta árvore, indicando as rotações necessárias;
- (b) retire os elementos {50, 95, 70, 60, 35} desta árvore, explicitando as rotações.

Grafos

Q13 - Desenhe os seguintes grafos:

- a) $G(V, E)$, onde $V = \{1, 2, 3, 4, 5, 6\}$ e $E = \{(2, 5), (6, 1), (5, 3), (2, 3)\}$.
- b) $G(V, E)$, onde $V = \{1, 2, 3, 4, 5, 6\}$ e $E = \{\{2, 5\}, \{6, 1\}, \{5, 3\}, \{2, 3\}\}$.

Q14 - Faça uma função para obter todos os nós adjacentes (vizinhos) a um nó do grafo, dado que o grafo é representado por uma matriz de adjacências.

Q15 - Faça uma função para obter todos os nós adjacentes (vizinhos) a um nó do grafo, dado que o grafo é representado por uma matriz de adjacências.

Q16 - Seja um grafo G cujos vértices são os inteiros de 1 a 8 e os vértices adjacentes a cada vértice são dados pela tabela abaixo:

Vértice	Vértices Adjacentes
1	2 3 4
2	1 3 4
3	1 2 4
4	1 2 3 6
5	6 7 8
6	4 5 7
7	5 6 8
8	5 7

- (a) Desenhe o grafo G .
- (b) Represente o grafo por meio de uma matriz de adjacência.
- (c) Represente o grafo por meio de uma lista de adjacência.