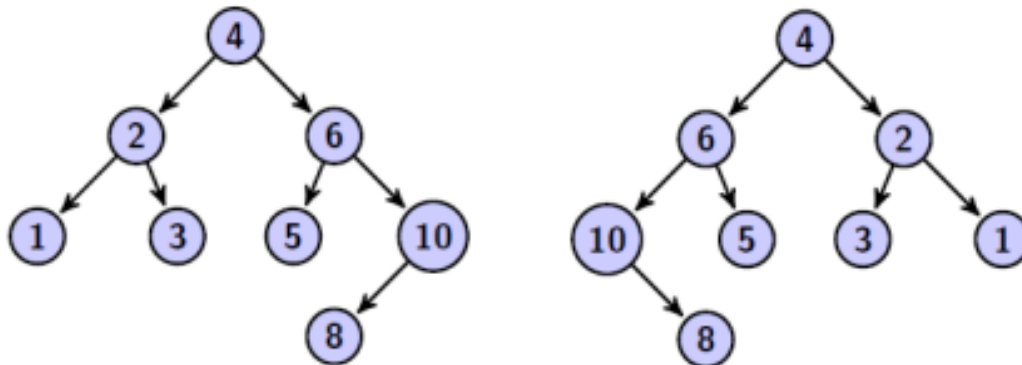


**2º Trabalho da disciplina de Estrutura de Dados I (noturno)**  
**Prof. Dr. Glauco Vitor Pedrosa**

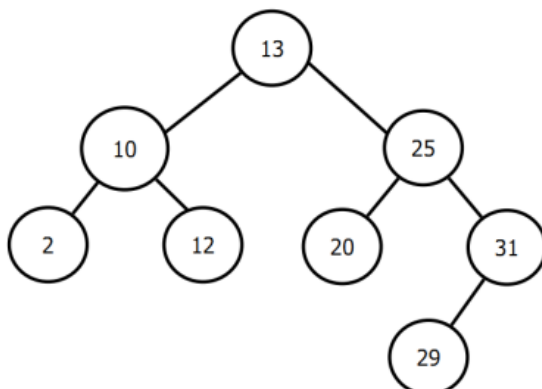
1 - Escreva uma função que obtém o espelho de uma árvore, ou seja, troca a subárvore direita pela subárvore esquerda de todos os nós da árvore



Ver arquivo: ArvBB-espelho.c

---

2 - O percurso em nível ou em largura em uma árvore é um percurso que visita, em ordem crescente, todos os nós de um nível antes de continuar a visita para o nível seguinte. Uma das formas de implementar é utilizar uma fila (FIFO) para guardar quais serão os próximos nós a serem visitados. Escreva um algoritmo para implementar o percurso em nível em uma árvore



**Seqüência de nós visitados**  
**13, 10, 25, 2, 12, 20, 31, 29**

Ver arquivo: ArvBB-largura.c

---

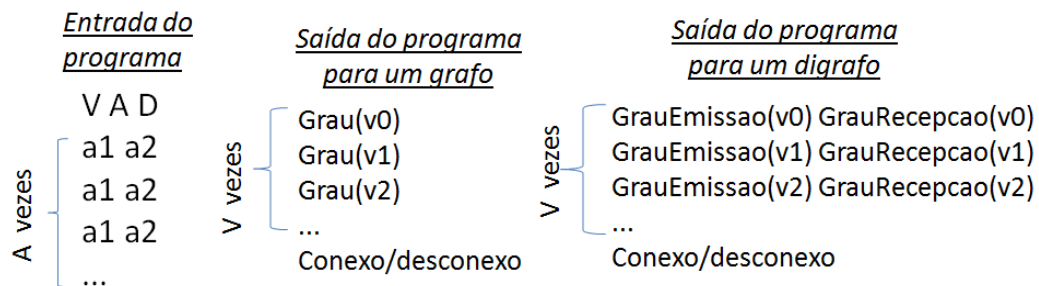
3- Implemente a função para remover um elemento de uma ABB (árvore binária de busca). Lembre-se, para remover um nó de uma ABB devemos considerar três casos: 1) nó sem filhos; 2) nó com um único filho; 3) nó com dois filhos.

Ver arquivo: ArvBB-remover.c

---

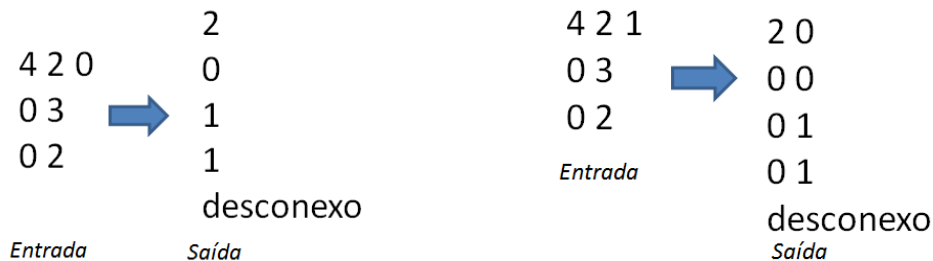
4 - Dado um grafo:

- calcular o grau de cada vértice
- verificar se ele é conexo ou desconexo



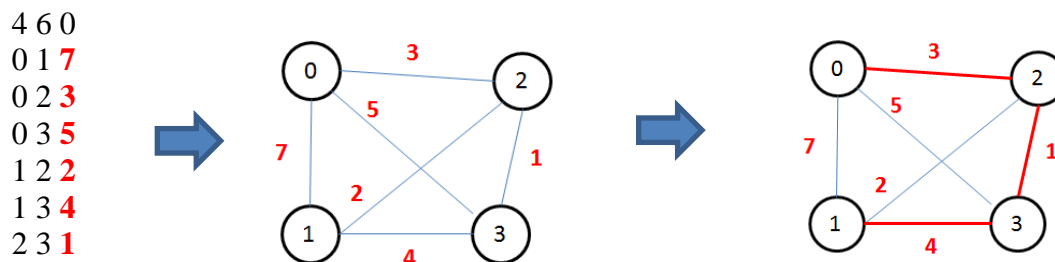
Em que, V é a quantidade de vértices do grafo, A a quantidade de arestas, a1 e a2 são arestas do grafo e D é 0 ou 1 (0 indica um grafo não-orientado e 1 indica um dígrafo)

Exemplo:



5) Implemente o algoritmo de PRIM para descobrir uma árvore geradora mínima em um grafo ponderado.

Exemplo de entrada do programa:



A entrada do programa é semelhante à entrada do programa anterior, exceto que o terceiro número de cada linha (a partir da segunda linha) se refere ao peso da aresta.

A saída deverá ser os vértices adicionados a cada passo do algoritmo, iniciando com o vértice 0:

Saída:

0 2 3 1

6) Dado um grafo, implemente uma função que realize uma busca em PROFUNDIDADE a partir do vértice 0.

Entrada do programa: igual a entrada do exercício 4

7) Dado um grafo, implemente uma função que realize uma busca em LARGURA a partir do vértice 0

Entrada do programa: igual a entrada do exercício 4