

Estruturas de Dados I (ED1)

Exercício 4

Prof. Carlos Astudillo

Percursos de árvores binárias de busca

Nesta tarefa, você deve implementar algumas operações a serem executadas em uma árvore binária de busca cujas chaves são valores inteiros positivos. Você deve considerar que nunca existirão chaves repetidas na árvore binária de busca.

As operações a serem implementadas são as seguintes:

- Inicializar a árvore binária de busca;
- Inserir uma chave na árvore binária de busca – esta operação consiste em determinar o local na árvore onde a chave deve ser inserida e adicionar neste local um novo nó contendo a chave.
- Imprimir o percurso em pré-ordem da árvore binária de busca;
- Imprimir o percurso em in-ordem da árvore binária de busca;
- Imprimir o percurso em pós-ordem da árvore binária de busca;
- Liberar a memória alocada para a árvore binária de busca.

A operação de liberar a memória alocada para a árvore binária de busca, **além de liberar** a memória alocada para a árvore binária de busca, deve **imprimir K** zeros seguidos, onde **K** é a quantidade de nós que a árvore binária de busca continha antes da sua memória ser liberada. Veja na Seção **Saída** uma descrição completa das impressões a serem feitas pelo seu programa.

Entrada

A primeira linha da entrada contém um inteiro **N** ($N \geq 2$), que indica a quantidade de operações a serem executadas na árvore binária de busca. Cada uma das **N** linhas seguintes contém um inteiro **O** ($1 \leq O \leq 6$) e, caso **O** seja igual a 2, **O** será seguido de um espaço em branco e um inteiro positivo **I**. Os inteiros **O** e **I** representam o seguinte:

- Se **O** é igual a 1, deve ser executada a operação de inicializar a árvore binária de busca;
- Se **O** é igual a 2, deve ser executada a operação de inserir a chave **I** na árvore binária de busca;
- Se **O** é igual a 3, deve ser executada a operação de imprimir o percurso em pré-ordem da árvore binária de busca;
- Se **O** é igual a 4, deve ser executada a operação de imprimir o percurso em in-ordem da árvore binária de busca;
- Se **O** é igual a 5, deve ser executada a operação de imprimir o percurso em pós-ordem da árvore binária de busca;
- Se **O** é igual a 6, deve ser executada a operação de liberar a memória alocada para a árvore binária de busca.

A operação de inicializar a árvore binária de busca (**O** igual a 1) sempre será a primeira operação especificada na entrada e ocorrerá apenas uma vez. A operação de liberar a memória alocada para a árvore binária de busca (**O** igual a 6) sempre será a última operação especificada na entrada e ocorrerá apenas uma vez.

Saída

A saída deve consistir no seguinte:

- Para cada execução de uma operação de imprimir um percurso da árvore binária de

busca, o seu programa deve imprimir o percurso correspondente no seguinte formato: |<int_1>||<int_2>|...|<int_k>|onde <int_1> é o inteiro contido no primeiro nó percorrido no percurso, <int_2> é o inteiro contido no segundo nó percorrido no percurso e assim por diante. Após imprimir o percurso, o seu programa deve passar para a próxima linha. A impressão do percurso e a passagem para a próxima linha podem ser feitas em diferentes pontos do seu programa.

- Quando for executada a operação de liberar a memória alocada para a árvore binária de busca, o seu programa deve imprimir **K** zeros seguidos, onde **K** é a quantidade de nós que a árvore binária de busca continha antes da sua memória ser liberada. Após realizar esta impressão, o seu programa deve passar para a próxima linha. A impressão dos zeros e a passagem para a próxima linha podem ser feitas em diferentes pontos do seu programa.

Exemplo de execução 1

```
7
1
2 4
2 3
2 5
3
4
6
|4||3||5|
|3||4||5|
000
```

Exemplo de execução 2

```
10
1
2 7
2 2
2 24
2 5
2 1
3
4
5
6
|7||2||1||5||24|
|1||2||5||7||24|
|1||5||2||24||7|
00000
```

| | |
|-------|--|
| Nota: | Textos em azul denotam dados de entrada do programa. |
| | Textos em vermelho denotam dados de saída do programa. |

Observações:

- Teste seu código compilando cada arquivo da seguinte forma: gcc -std=c99 -pedantic -Wall -lm -o percursos *.c.