

Prof. Hebert Coelho
Profa Nádia Félix

Conteúdo

1	Soma Árvore Binária de Busca	2
2	Árvore Binária de Busca	3
3	Tree Sort	5
4	Busca em ABB	6

1 Soma Árvore Binária de Busca



(+++)

Faça um programa para inserir elementos inteiros em uma árvore binária de busca. Após apresentar a soma dos elementos que são folhas seguido da soma dos elementos internos.

Entrada

A primeira linha contém os elementos inteiros separados por espaço a serem inseridos na árvore.

Saída

Você deverá imprimir dois valores inteiros separados por espaço, o primeiro com a soma dos elementos que são folhas, o segundo com a soma dos elementos internos (não são folhas).

Exemplos

Entrada	Saída
1 2 3 4 5 6 7 8 9 10	10 45

Entrada	Saída
5 3 7 2 4 6 8	20 15

2 Árvore Binária de Busca



(+++)

A árvore binária de busca (ABB) é um dos assuntos mais importantes em Estruturas de Dados. Abaixo são apresentados a estrutura de um nó e uma forma de criar uma raiz nula em linguagem C.

Estrutura de um nó:

```
typedef struct cel{  
int chave;  
struct cel *esq;  
struct cel *dir;  
} no;  
typedef no *arvore;
```

Criação da árvore vazia:

```
arvore raiz = NULL;
```

Algoritmo de inserção:

Recebe a raiz “p” de uma árvore de busca e uma chave “k”. Caso não exista um nó com chave “k”, cria o nó com chave “k” e insere na árvore.

Algoritmo de remoção de uma ABB:

- 1) A remoção de um nó terminal (folha) é direta;
- 2) Ao se remover um nó de chave x, com apenas 1 descendente, o ancestral de x será o ancestral do descendente de x;
- 3) Na remoção de um nó de chave x, com mais de um descendente, o nó x será substituído pelo menor elemento da subárvore direita.

Implemente um programa que faça N inserções e/ou remoções utilizando a estrutura e algoritmos acima e ao final escrever na tela a altura da árvore binária de busca.

Entrada

A primeira linha de entrada contém um único inteiro positivo N ($1 < N < 100000$). Este é o número de linhas de entrada que vem logo a seguir. As próximas N linhas conterão, cada uma delas, 2 valores inteiros. O primeiro valor é 1 para a inserção ou 2 para remoção, o segundo valor é a chave a ser inserida ou removida.

Saída

Ao final o seu programa deverá imprimir a altura da árvore.

Exemplos

Entrada	Saída
5 1 4 1 3 2 1 2 2 1 5	2

Entrada	Saída
10 1 5 1 8 1 3 1 4 1 2 1 1 1 0 1 7 1 9 2 3	5

Entrada	Saída
7 1 1 1 2 1 3 1 4 1 0 1 5 2 3	4

3 Tree Sort



(+++)

Um método de ordenação conhecido como TreeSort é basicamente construir uma árvore binária de busca (ABB) e percorrer tal árvore em ordem simétrica apresentando as chaves. Implementar o método TreeSort usando uma ABB. A estrutura a ser armazenadas na ABB é:

Entrada

A primeira linha contém um inteiro $N < 1000000$ com o número de linhas após a primeira linha. As próximas N linhas contém 2 valores inteiros com no máximo 4 dígitos. O primeiro valor é a chave e segundo valor é o número de matrícula.

Saída

Seu programa deve imprimir um número de matrícula em cada linha, na sequência da menor chave para a maior.

Exemplo

Entrada	Saída
10 5 1178 4 1287 3 1365 8 1312 9 1410 1 4161 0 3425 2 6543 6 9871 7 2134	3425 4161 6543 1365 1287 1178 9871 2134 1312 1410

Entrada	Saída
5 5 3251 19 1410 10 4161 1 3425 6 1583	3425 3251 1583 4161 1410

4 Busca em ABB



(+++)

Suponha que se queira pesquisar a chave 287 em uma árvore binária de pesquisa com chaves entre 1 e 1000. Durante uma pesquisa como essa, uma sequência de chaves é examinada. Cada sequência abaixo é uma suposta sequência de chaves examinadas em uma busca da chave 287.

1. 7, 342, 199, 201, 310, 258, 287
2. 110, 132, 133, 156, 289, 288, 287
3. 252, 266, 271, 294, 295, 289, 287
4. 715, 112, 530, 249, 406, 234, 287

Caso você desenhe a árvore binária de pesquisa relativa a cada item acima, verá que apenas as duas primeiras sequências serão válidas. Crie um programa que construa uma árvore binária de pesquisa através de inserções da sequência de chaves dada, realize uma busca por uma chave específica, e a partir dos dados da busca diga se a sequência de chaves dada é válida ou não.

Entrada

A primeira linha contém a chave a ser procurada. A segunda contém sequência de chaves separadas por espaço a ser examinada.

Saída

Você deverá imprimir V quando a sequência for válida e I quando a sequência for inválida.

Exemplo

Entrada	Saída
287 7 342 199 201 310 258 287	V

Entrada	Saída
287 252 266 271 294 295 289 287	I