

## Exercício 11 - Verificar se é uma ABB

### Descrição

Elabore uma função que verifique se uma árvore binária é uma Árvore Binária de Busca.

### Entrada

A primeira linha da entrada contém um inteiro  $N$ , indicando a quantidade de elementos que serão inseridos na árvore binária, as próximas  $N$  linhas contém 3 inteiros, representando o elemento que será inserido, o lado (direito ou esquerdo) e a chave pai do elemento que será inserido.

### Saída

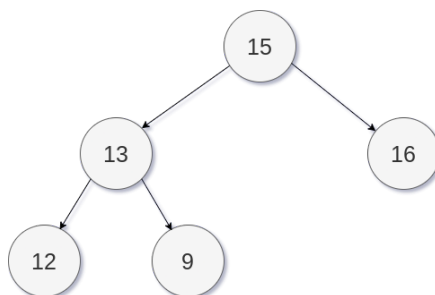
Imprima “É uma ABB.” caso seja uma ABB e “Não é uma ABB.” caso contrário.

Exemplo de Entrada	Exemplo de Saída
--------------------	------------------

```
5
15 0 15
13 0 15
16 1 15
12 0 13
9 1 13
```

Não é uma ABB.

### Representação da Árvore Binária do Exemplo:



## Observações:

- Somente as bibliotecas *stdio.h*, *stdlib.h* e *stdbool.h* podem ser utilizadas.
- O exercício deve ser resolvido utilizando o TAD AB;
- Deve ser elaborada uma função chamada *is\_abb* de acordo com a declaração presente no arquivo *ab.h* (com o mesmo nome, entrada e saída);

```
bool is_abb(AB *T);
```

- A solução para o problema deve ser elaborado em um arquivo *.c* chamado *is\_abb.c*
- O arquivo que será submetido ao RunCodes deve ser um *.zip* apenas com a solução *is\_abb.c*

## Definição de tipos:

```
typedef struct no_ NO;

struct no_{
    ITEM *item;
    NO *esq;
    NO *dir;
};

struct ab_{
    NO *raiz;
    int profundidade;
};
```