

Árvores: Busca em Profundidade Em-Ordem

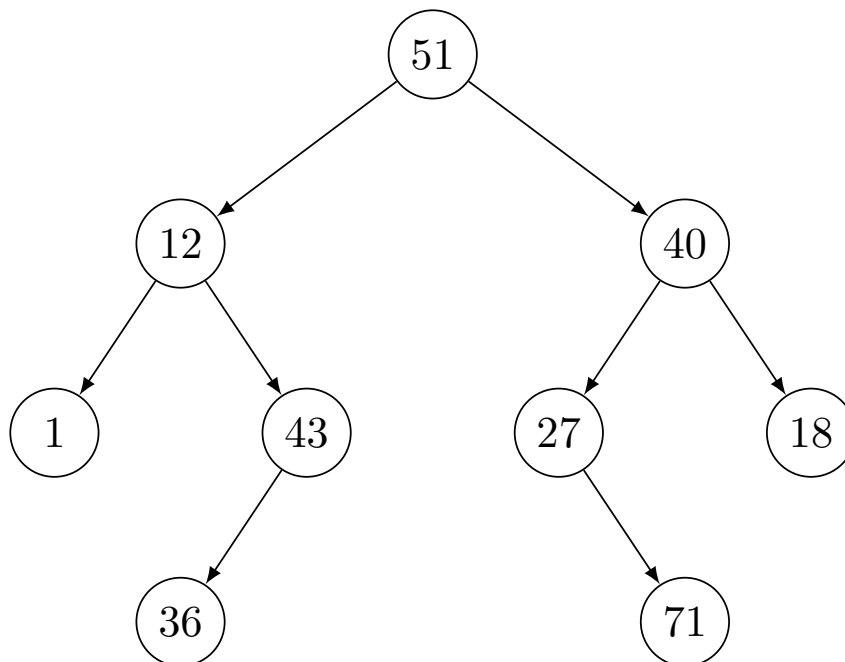
Limite de tempo: 1s
Limite de memória: 256MB

Autor: Daniel Saad Nogueira Nunes

Implemente o seguinte procedimento para realizar uma busca em profundidade **em-ordem**. Para cada nó visitado na ordem correta, deve-se imprimir o seu valor em uma linha.

```
voidk dfs_in(tree_t *t);
```

Por exemplo, para a árvore abaixo, a ordem de impressão dos valores seria: 1, 12, 36, 43, 51, 27, 71, 40, 18.



Considere a seguinte definição de árvore binária.

```
typedef struct tree_node_t{
    int data;
    struct tree_node_t *left;
    struct tree_node_t *right;
} tree_node_t;

typedef struct tree_t{
    tree_node_t *root;
} tree_t;
```

Notas

Só é necessário implementar a função pedida no enunciado. Não é necessário realizar a leitura dos dados ou escrever a função `main`. Funções auxiliares, que tenham a ver apenas com a função solicitada, podem ser implementadas, caso deseje.

Se estiver programando em C ou C++ é necessário incluir o cabeçalho “**grader.h**” na sua solução:

```
#include "grader.h"
```