

### Árvores Binárias

Considere a seguinte definição de Árvore Binária:

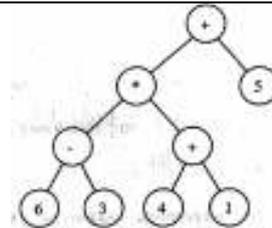
```
struct arv {
    char info;
    struct arv* esq;
    struct arv* dir;
};
typedef struct arv Arv;
```

**1)** Implemente uma função que retorne a quantidade de folhas de uma árvore binária. Essa função deve obedecer ao protótipo: **int folhas (Arv\* a);**

**2)** Implemente uma função que compare se duas árvores binárias são iguais. Essa função deve obedecer ao protótipo: **int igual (Arv\* a, Arv\* b);**

**3)** Considere uma árvore binária que representa expressões. Por exemplo, a expressão  $(6-3)*(4+1)+5$  é representada pela árvore binária ilustrada na figura abaixo. As folhas da árvore armazenam operandos e os nós internos operadores. Se avaliada, esta expressão resulta no valor 20. Considere a existência do tipo usado para representar árvores binárias de expressões.

```
struct arv {
    char op;
    float valor;
    struct arv *esq, *dir;
}
typedef struct arv Arv;
```



O campo *valor* é usado apenas pelas folhas e o campo *op* pelos nós internos. Escreva funções que, dada a raiz de uma árvore binária de expressões (pode-se considerar que a árvore nunca está vazia):

**(a)** Imprima a expressão em notação pós-fixada. Para exemplo acima, imprimiria: 63-41+\*5+. O protótipo da função deve ser: **void imprime (Arv\* a)**

**(b)** Retorne o valor correspondente à avaliação da expressão (para o exemplo acima, retornaria 20). O protótipo da função deve ser: **float avalia (Arv\* a)**

**4)** Considere uma *árvore binária de busca* que armazena valores inteiros. Nesta estrutura, pode ocorrer repetições de um mesmo valor. Assim, os valores associados aos nós da subárvore da esquerda são menores que o valor associado à raiz, e os valores associados à subárvore da direita são maiores ou iguais. O tipo que representa o nó da árvore é dados por:

```
struct arv {
    int info;
    struct arv* esq;
    struct arv* dir;
};
typedef struct arv Arv;
```

**(a)** Escreva uma função que retorne o número de ocorrências de um dado valor *x* na árvore. A função deve tirar proveito da ordenação da árvore e obedecer ao seguinte protótipo: **int ocorrencias (Arv\* a, int)**

**(b)** Escreva uma função que imprima os valores associados às folhas em ordem não crescente (isto é, do maior para o menor, podendo haver repetições). A função deve obedecer ao seguinte protótipo: **int imprime\_folhas (Arv\* a)**