

9.1ª lista de exercícios de Programação e Estruturas de Dados II

Problema 1

Implemente versões diferentes da função imprime, percorrendo a árvore em ordem simétrica e em pré/pós-ordem. Verifique o resultado da aplicação das duas funções na árvore da Figura 13.5.

Problema 2

Implemente uma função que retorne a quantidade de folhas de uma árvore binária. Essa função deve obedecer ao protótipo:

```
int folhas (Arv* a);
```

Problema 3

Implemente uma função que compare se duas árvores binárias são iguais. Essa função deve obedecer ao protótipo:

```
Arv* igual (Arv* a, Arv* b);
```

Problema 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 sub-árvore à esquerda são menores que o valor associado à raiz e os valores dos nós da sub-árvore à direita são maiores ou iguais. O tipo que representa um nó da árvore é dado por:

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

a) Implemente a implantação de uma árvore binária de busca.

b) Escreva uma função que retome 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 ocorrenciasx (Arv* a, int x);
```

c) Escreva uma função que imprima os valores associados às folhas da árvore 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);
```

Problema 5

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 os operandos e os nós internos operadores. Se avaliada, esta expressão resulta no valor 20.

Considere a existência do tipo abaixo usado para representar árvores binárias de expressões.

```

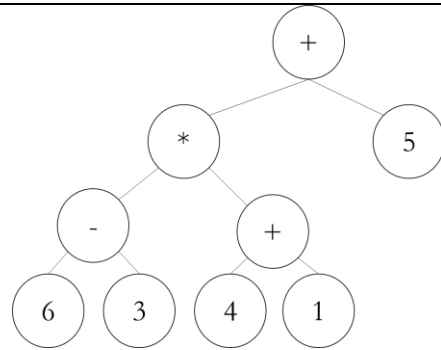
struct arv {
    char op; /* + - * / */
    float valor;
    struct arv *esq, *dir;
} ;

```

```

typedef struct arv Arv;

```



onde 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 será vazia):

a) Imprima a expressão em notação pós-fixada. Para exemplo acima, imprimiria: `6 3 - 4 1 + * 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);

```