

# Programação 2

Segunda lista de exercícios — 12/06/2001

## Fila

1. Defina rotinas para:

- inserir (recursivamente) um número inteiro em uma fila;
- remover o próximo número inteiro de uma fila;
- retornar o próximo número inteiro de uma fila.

Considere que a fila foi definida da seguinte maneira:

```
reg FilaInt { int dado, FilaInt prox } ;
```

## Pilha

1. Defina rotinas para:

- inserir um número inteiro em uma pilha;
- remover o próximo número inteiro de uma pilha;
- retornar o próximo número inteiro de uma pilha.

Considere que a pilha foi definida da seguinte maneira:

```
reg PilhaInt { int dado, PilhaInt prox } ;
```

## Recursividade

1. Defina procedimentos recursivos para:

- inserir um número inteiro no final de uma lista;
- retornar o número inteiro do final de uma lista.

Considere que a listafoi definida da seguinte maneira:

```
reg ListaInt { int dado, ListaInt prox } ;
```

## Árvore

1. Defina uma rotina recursiva para inserir um número inteiro em uma árvore binária de busca.
2. Defina uma função recursiva que diz se um número inteiro está ou não em uma árvore binária de busca.

Considere que a árvore foi definida da seguinte maneira:

```
reg ArvoreInt { ArvoreInt esquerda, int dado, ArvoreInt direita } ;
```

## Algoritmos de ordenação

1. Implemente procedimentos (rotinas) para ordenar um array de inteiros com os seguintes algoritmos de ordenação:

- BubbleSort;
- SelectionSort;
- InsertionSort.