

## 1º Lista de Exercícios – Pilha, Fila e Lista (Alocação Estática de Memória)

1. Tendo como base a implementação de fila realizada em sala de aula, implementar uma pilha de inteiros com alocação estática de memória. A implementação deve obedecer a seguinte interface. Implementar uma classe para teste.

```
public interface Pilha {  
  
    /**  
     * Metodos obrigatorios  
     */  
    public Integer insereTopo(Integer valor);  
    public Integer removeTopo();  
  
    /**  
     * Metodos secundarios  
     */  
  
    public boolean pilhaCheia();  
    public boolean pilhaVazia();  
  
    public void imprimePilha();  
}
```

2. Cite duas desvantagens da implementação com alocação sequencial estática resolvidas com a alocação dinâmica.
3. Indique em que situações é melhor utilizar uma lista encadeada ao invés de uma lista usando vetor.
4. Descreva a saída resultante da seguinte série de operações de pilha: push(5), push(3), pop(), push(2), push(8), pop(), pop(), push(9), push(1), pop(), push(7), push(6), pop(), pop(), push(4), pop(), pop().
5. Descreva a saída resultante da seguinte série de operações de fila: enqueue(5), enqueue(3), dequeue(), enqueue(2), enqueue(8), dequeue(), dequeue(), enqueue(9), enqueue(1), dequeue(), enqueue(7), enqueue(6), dequeue(), dequeue(), enqueue(4), dequeue(), dequeue().
6. Implementar uma lista utilizando alocação estática de memória. A implementação deve obedecer a seguinte interface.

```

public interface Lista<E> {

    /** Operações básicas */
    public abstract E set(int pos, E e);

    public abstract void addPrimeiro(E e);
    public abstract void addUltimo(E e);

    public abstract void addAntes(int pos, E e);
    public abstract void addApos(int pos, E e);

    public abstract void remove(E e);
    public abstract void remove(int pos);

    /** Operações auxiliares */
    public abstract E first();
    public abstract E last();

    public abstract E anterior(int pos);
    public abstract E posterior(int pos);
}

```

7. Escreva uma operação para remover de uma lista encadeada todos os elementos que contém y.
8. Suponha que você deseja armazenar o Ranking das 10 maiores pontuações para um determinado jogo. Esse ranking deve conter o nome e pontuação.

```

public class PontuacaoJogador {

    private String nome;
    private int pontuacao;

}

```

A lista deverá exibir a pontuação dos jogadores de forma ordenada, como na figura abaixo. Você deve implementar essa estrutura de dados, de forma que a inserção seja de maneira ordenada e na remoção, os elementos posteriores ao elemento removido sejam deslocados uma posição para a esquerda. Implemente também uma classe de teste.

