

Exercícios de revisão P1

1) Utilizando recursividade, fazer um algoritmo que apresente quantos dígitos tem um número. Apresente o ponto de parada e a relação para a chamada do próximo passo. Implementar o algoritmo em Java.

2) Considerando o seguinte vetor:

89	105	4	-32	7	44	-2	0	50	-10
----	-----	---	-----	---	----	----	---	----	-----

Responder:

- Quantas rodadas Bubble Sort são necessárias para que o vetor já esteja ordenado ?
- Qual a ordem de aparição dos índices dos pivôs fixos na ordenação Quick Sort ?

3) Considerando as implementações de fila conforme a biblioteca Filaint.jar gerada em aula, simular, como um teste de mesa, apresentando como ficam a saída em console ao final da seguinte implementação:

```
package controller;
import br.com.leandrocolevati.filaint.Fila;
public class OpController {
    public OpController() {
        super();
    }
    public void op(int[] vetor) {
        Fila f = new Fila();
        for (int i : vetor) {
            if (i % 2 == 1) {
                f.insert(i);
            } else {
                if (!f.isEmpty()) {
                    int valor = f.remove() + i;
                    System.out.println(valor);
                }
            }
        }
        System.out.println(f.toString());
    }
}
```

```
package view;
import controller.OpController;
public class Principal {
    public static void main(String[] args) {
        int[] vetor = {145,285,220,172,204,149,112,2,96,29,180,271,290,121,117,221,126};
        OpController opCont = new OpController();
        opCont.op(vetor);
    }
}
```

4) Considerando as implementações de fila conforme a biblioteca Pilhaint.jar gerada em aula, simular, como um teste de mesa, apresentando como ficam a saída em console ao final da seguinte implementação. Apresente também a estrutura interna da pilha caso a aplicação finalize com valores na pilha:

```
package controller;
import br.com.leandrocolevati.filaint.Fila;
public class OpController {
    public OpController() {
        super();
    }
    public void op(int[] vetor) throws Exception {
        Pilha p = new Pilha();
        for (int i : vetor) {
            if (i % 2 == 1) {
                p.push(i);
            } else {
                if (!p.isEmpty()) {
                    int valor = p.pop() + i;
                    System.out.println(valor);
                }
            }
        }
    }
}
```

```
package view;
import controller.OpController;
public class Principal {
    public static void main(String[] args) {
        int[] vetor = {145,285,220,172,204,149,112,2,96,29,180,271,290,121,117,221,126};
        OpController opCont = new OpController();
        try {
            opCont.op(vetor);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```