

ESQUEMA AULA PRÁTICA 6

□ A Classe Vector

1 - Defina e desenvolva em JAVA a classe FilaDeEspera que deverá manter os nomes dos pacientes em espera no consultório médico do Dr. Caharoo Masbom. Para guardar os nomes dos pacientes use um Vector de Strings (Objeto do tipo `java.util.Vector<String>`). Cada instância de classe FilaDeEspera deverá ser capaz de responder adequadamente a um conjunto de mensagens correspondentes às operações de:

- Consulta do número de pessoas em espera (comprimento actual da fila).
- Consulta do nome do primeiro paciente da fila.
- Chegada de um novo paciente (inserção de um elemento na cauda da fila).
- Atendimento de um paciente (remoção do primeiro elemento da fila).
- Mostrar o nome de todos os pacientes em espera. (método `toString`).
- Comparar dois objectos (método boolean `equals (Object o)`).
- Criar um cópia do objecto (método `Object clone()`).

2 – Construa uma classe de teste para verificar a classe anterior.

Para o exercício 3 considere a classe Jogador que construiu na folha prática 4.

3 - Uma equipa tem como atributos o nome da equipa e um “Vector” (objecto do tipo `java.util.Vector`) de jogadores onde são armazenados os jogadores da equipa.

A classe Equipa terá um construtor com o nome da equipa como parâmetro e cada instância deverá poder responder às seguintes mensagens:

- inserir um novo jogador na equipa;
- remover um jogador da equipa dada a sua posição no Vector de jogadores;
- dar a conhecer o número de jogadores da equipa
- dado o nome de um jogador, verificar este pertence ou não à equipa;
- devolver o nome do jogador que marcou mais golos no campeonato.
- mostrar sob a forma de texto o estado de um objecto do tipo equipa (método `toString`)
- Comparar dois objectos (método boolean `equals (Object o)`).
- Criar um cópia do objecto (método `Object clone()`).

4 - Construa uma classe de Teste em que verifique a correcção dos métodos da classe Equipa.

API da classe `java.util.Vector`:

- `Vector()` // construtor vector vazio, dimensão inicial zero.
- `Vector(int capacidInicial)` // construtor vector vazio, com dimensão inicial.
- `void addElement(Object elemento)`
 - // adiciona o elemento especificado ao final do vector.
- `void insertElementAt(Object obj, int indice)`
 - // insere o elemento especificado na posição indice.
- `void removeElementAt(int indice)` // remove o elemento na posição indice.
- `void setElementAt(Object obj, int indice)`
 - // substitui o elemento da posição indice pelo objecto dado.
- `Object elementAt(int indice)` // devolve o componente presente no indice.
- `void clear()` // remove todos os elementos do vector.
- `Object clone()` // devolve uma cópia do vector.
- `Boolean equals (Object elemento)`
 - // compara o Vector com o Vector dado, elemento.
- `boolean contains(Object elemento)`
 - // verifica se o objecto especificado é um componente deste vector
- `Object firstElement()` // devolve o primeiro componente (indice 0) do vector.
- `Object lastElement()` // devolve o último componente do vector.
- `boolean isEmpty()` // verifica se o vector não tem componentes
- `int size()` // devolve a dimensão actual.
- ...