

ESQUEMA AULA PRÁTICA 5

□ Classes e Objetos

□ A classe String

1 – Construa a classe *Disciplina* esquematizada abaixo:

```
public class Disciplina {
    private int codigo;
    private String nome;
    private String prof;

    public Disciplina ( ) {
        codigo = 0;
        nome = "";
        prof = "";
    }
    public Disciplina (int codigo, String nome){
        this.codigo = codigo;
        this.nome = nome;
        this.prof = "";
    }
    ...
}
```

a) Para a classe *Disciplina* construa o método “toString”.

b) Para a classe *Disciplina* construa os métodos de consulta (getters) e de modificação (setters) para todos os atributos.

c) Para a classe *Disciplina* construa um método que verifique se o nome da disciplina contém uma substring dada pelo utilizador.

d) Para a classe *Disciplina* construa um método que verifique se o seu código está num dado intervalo do tipo [inicio, fim[; O método deve receber como parâmetros os valores de início e fim e como resultado deve devolver **true** se a código da disciplina está nesse intervalo, e **false** caso contrário.

2 – Construa uma classe de teste para a classe *Disciplina*.

3 – Suponha uma classe que representa as 3 cadeiras mais críticas de um curso.

a) Construa a classe *CadeirasCriticas*, definindo os atributos *curso* (uma *String*) que representa o nome do curso, e *disciplinas* (um array com 3 *Strings*) que representa as 3

disciplinas críticas desse curso. Defina os atributos, um construtor que receba como parâmetro o nome do curso, construa os getters os setters e o método toString.

b) Construa uma classe para testar os **métodos** da classe CadeirasCriticas. Deverá:

- declarar pelo menos dois objetos do tipo cadeiras críticas, cc1 e cc2;
- instanciar os objetos;
- atribuir valores a todos os atributos dos objeto;
- modificar esses valores e mostrar o conteúdo de cada objeto.

c) Teste o método setDisciplinas, da seguinte forma:

- declarar no main um array de Strings com dimensão 3 (de nome novasCriticas);
- inicialize o array com valores à sua escolha;
- use esse array para modificar o valor da lista de cadeiras criticas do objeto cc1;
- invoque o método toString no objeto cc1 para ver o conteúdo do objeto;
- atribua novos valores ao array novasCriticas;
- invoque o método toString e analise o que aconteceu? O seu método setDisciplinas estava mesmo correto? Se não estava, corrija-o.

4 – Construa uma classe de Teste para estudar a classe **String** predefinida na linguagem Java.

Teste cada um dos métodos seguintes:

```
char charAt(int);  
int length();  
int indexOf(String);  
String concat( String);  
String substring(int);  
int compareTo (String);
```

5 - Construa um programa onde deve definir uma String com um dos textos abaixo, à sua escolha:

“The competent programmer is fully aware of the limited size of his own skull. He therefore approaches his task with full humility, and avoids clever tricks like the plague.”

[Edsger Dijkstra, 1972](#)

“A programmer – someone who creates algorithms and codes them up – is a minor god, creating universes at will. You could even say that the God of Genesis himself is a programmer: language, not manipulation, is his tool of creation. Words become worlds. Today, sitting on the couch with your laptop, you too can be a god. Imagine a universe and make it real.”

[Pedro Domingos, 2016](#)

- O programa deverá:

- a) Contar quantos caracteres tem a String (incluindo espaços);
- b) Contar quantos caracteres (sem contar os espaços) tem a String;
- c) Contar quantas palavras tem a String;
- d) Contar o número de vezes que ocorreu cada uma das letras do alfabeto na String.
- e) Dada uma palavra verificar se ela pertence ao texto e em que posição está.

Para estudar mas tarde:

6 – Suponha agora o exercício 3, mas em vez de o array de disciplinas ser um array de Strings, suponha que era um array de objetos do tipo Disciplinas. Reconstrua a classe. Depois de construir a classe, teste o método `setDisciplinas` como sugerido no exercício 3.