ESQUEMA AULA PRÁTICA 8

- Herança, Polimorfismo e Ligação em Java

Podem definir-se novas classes a partir de classes já existentes. A nova classe, classe derivada ou subclasse, vai herdar as características e funcionalidades da classe base ou superclasse, podendo redefinir funções e/ou adicionar novos membros. As restrições de acesso são estendidas à subclasse e seus clientes. Enquanto os membros da classe derivada não têm acesso aos membros privados da classe base, os membros públicos da classe base são automaticamente membros da classe derivada.

- Suponhamos a classe **C1**, definida abaixo, que contém dois métodos **m1 e m2**:

```
public class C1 {
   public void m1() {
     System.out.println ("Método 1 classe C1");
   }
   public void m2() {
     System.out.println ("Método 2 classe C1");
   }
}
```

- Suponha também uma subclasse de C1, a classe **C2** em que o método **m2** é redefinido:

```
public class C2 extends C1{
    public void m2() {
     System.out.println ("Método 2 classe C2 ");
    }
}
```

Os membros m1 e m2 são métodos polimórficos que podem ser aplicados a vários tipos de objectos: m1 ilustra o polimorfismo de inclusão, pois pode ser invocado em objectos do tipo C1 e qualquer subtipo deste, enquanto m2 ilustra o polimorfismo de sobreposição ("overriding"). O tipo de objecto em que m2 for invocado determinará qual a implementação do método que vai ser executada.

1. Qual será o output da execução do seguinte segmento de código?

```
public static void main (String str[]) {
    C1 o1= new C1();
    C2 o2= new C2();
    o2.m1();
    o2.m2();
    o1.m2();
}
```

- Implemente o exemplo para verificar a sua resposta.

2. Suponha agora que definíamos um método, f, que tem como parâmetro um objecto do tipo C1,

```
void f (C1 x1){
    x1.m2();
  }
```

e que invocamos a função, respectivamente com um objecto da classe C1 e classe C2:

```
f (o1);
f (o2);
```

- Qual acha que seria o output dessa execução?

Faça dois testes:

a) Primeiro, declare a função dentro da classe de teste e invoque-a no método main.

b) Segundo, defina uma classe C com o método f, e seguidamente na classe de teste declare um objecto da classe C, e invoque o método f sobre esse objecto.

- O que pode concluir quanto ao mecanismo de ligação da linguagem, é estático ou dinâmico?

3. Programe a classe FilaDeEspera como uma especialização (subclasse) da classe Vector.

4. Simplificando, podemos afirmar que, uma pessoa é alguém de quem sabemos o nome, o sexo e a nacionalidade. Programe a classe `Pessoa` com os construtores (entre eles implemente o construtor não parametrizado – qual a utilidade? – e o *construtor por cópia*), selectores e modificadores que considerar necessários bem como os métodos públicos `toString`, `clone` e `equals` (ainda que os considere pouco naturais para uma pessoa terão interesse do ponto de vista da programação!).

5. Programe a classe `Autor` de obra literária. Um autor é uma pessoa sobre a qual será interessante saber o ano de nascimento e falecimento (caso já tenha falecido), a sua nacionalidade bem como o prémio literário mais importante que conquistou, se detivermos esta informação.

6. Programe a classe `Musico`. Um músico pode ser um indivíduo ou uma banda de que interessa manter uma descrição e o seu estilo musical dominante.

7. Programe a classe `Amigo`. Um amigo é uma pessoa de quem sabemos a data de nascimento, o ano em que o conhecemos, um contacto, o nível de amizade que por ela nutrimos e ainda o "parceiro" com quem normalmente "anda".