

**MC322/MC336**  
Segundo semestre de 2015

**Laboratório 1**

**Professor:** Fábio Luiz Usberti (fusberty@ic.unicamp.br)

**PED:** Rafael Arakaki (rafaelkendyarakaki@gmail.com)

---

## 1 Objetivo

O objetivo desta atividade consiste na familiarização com o Ambiente de Desenvolvimento Integrado (IDE, *Integrated Development Environment*) chamado Eclipse<sup>1</sup> para a linguagem de programação Java<sup>2</sup>.

## 2 Atividade

Nesta atividade o principal foco será a familiarização com o Eclipse e a programação de uma classe chamada **Carta**. A primeira tarefa será configurar o ambiente com a criação de um novo projeto e de uma nova classe para então programar.

Os seguintes passos podem ser tomados para a criação do projeto e da classe:

1. Abra o Eclipse.
2. Crie um novo projeto (File -> New -> Java Project).
3. Crie uma nova classe no projeto (Botão direito no projeto -> New -> Class).
4. A classe deve se chamar **Carta**. Certifique-se de que a opção de criação da função main está selecionada.
5. Programe a classe.
6. Para executar o programa e ver o resultado clique no botão “Run”.

## 3 Classe Carta

A classe Carta deste laboratório é baseada em um jogo de cartas de computador chamado Hearthstone<sup>3</sup> ©, neste jogo existem cartas do tipo *lacaio* que possuem valores como ataque e vida diferentes para cada carta.

A classe Carta deve ter os seguintes atributos:

- Nome (cadeia de caracteres - *String*)
- Ataque (número inteiro)
- Vida (número inteiro)

---

<sup>1</sup><https://eclipse.org>

<sup>2</sup><https://www.java.com>

<sup>3</sup><http://us.battle.net/hearthstone/pt>

- Mana (número inteiro)

Além disso a classe carta deve conter uma função construtora, a função construtora recebe como argumentos os atributos para inicializar o objeto. Para ilustrar esta função melhor, veja o exemplo abaixo, realizado considerando uma classe **Carro** com os atributos nome, marca e ano:

```

1 // — Dentro da classe Carro
2 public Carro(String nome, String marca, int ano) {
3     this.nome = nome;
4     this.marca = marca;
5     this.ano = ano;
6 }
7 public String getNome() {
8     return nome;
9 }
10 public void setNome(String nome) {
11     this.nome = nome;
12 }
13 // outras funcoes...
```

construtor.java

Também é necessário que a classe Carta possua uma função **toString()** que devolve uma String contendo uma descrição geral dos atributos do objeto carta. Veja o exemplo do Carro da seguir:

```

1 public class Carro {
2
3     private String nome;
4     private String marca;
5     private int ano;
6
7     @Override
8     public String toString() {
9         String out = "";
10        out = out + "Nome= " + getNome();
11        out = out + ", Marca= " + getMarca();
12        out = out + ", Ano= " + getAno();
13        return out;
14    }
15
16    // Outras funcoes abaixo (get, set, ...)
17 }
18 }
```

Carro.java

Observe que são utilizados os métodos de get e set, que são métodos muito utilizados na linguagem Java para proteger os atributos dos objetos. É necessário programar estes métodos antes de utilizá-los, logo para cada atributo da classe Carta deve existir um método get e set correspondente.

Por fim o programa deve executar na função main a impressão dos atributos de alguns objetos instanciados da classe Carta. Para isso, instancie alguns objetos do tipo Carta com valores de atributos quaisquer conforme sua imaginação. Veja o exemplo da classe Carro a seguir:

```

1 // — dentro da classe Carro
2 public static void main(String[] args) {
3     // Imprime os atributos de alguns carros que foram projetados no Brasil
4     Carro carroDoMeuPai = new Carro("Gol", "Volkswagen", 1980);
5     Carro meuCarro = new Carro("Novo Uno", "Fiat", 2010);
6     System.out.println(carroDoMeuPai);
7     System.out.println(meuCarro);
8 }
```

8

}

print.java

Observe que para imprimir dados dos objetos da classe Carta, a função `println()` utiliza o método `toString()` que você implementou, esta função existe para todos os objetos e é uma característica da linguagem Java que pode ajudar o(a) programador(a) no processo de depuração do código.

## 4 Submissão

Para submeter desta atividade utilize a página da disciplina no Ensino Aberto. Utilize o recurso de portfólio para submeter a atividade. Para isso, crie uma subpasta dentro de seu portfólio com o nome **Lab1** e dentro desta subpasta submeta o arquivo Carta.java com a sua implementação da classe Carta.