

**Universidade Federal de Campina Grande – UFCG**  
**Centro de Engenharia Elétrica e Informática – CEEI**  
**Departamento de Sistemas e Computação – DSC**

Professores: Livia Campos (turma1)  
Reinaldo Gomes (turma2)  
Nazareno Andrade (turma 3)

Disciplina: Laboratório de Programação 2  
Período: 2011.1

## Laboratório Guiado 01

---

Neste laboratório iremos experimentar a edição e compilação de programas simples em Java (explorando suas estruturas básicas) e preparar o ambiente de programação para as próximas aulas.

### **Importante:**

- ◆ Data de entrega: 17/02/2011, até 12h (ou 18/02/2011, até 10h, turma 3);
- ◆ Certifique-se de que seus programas não têm erros de compilação;
- ◆ Não copie o programa do seu vizinho. Se tiver dúvida converse com o professor ou com um monitor;
- ◆ Antes da definição de cada classe escreva o seu nome em comentário (`/*Aluno: <seu nome>*/`);
- ◆ Salve as classes criadas em um arquivo zip chamado labGuiado01-<seuNome>.zip (ou .tgz) e o envie para [juiz.lp2@gmail.com](mailto:juiz.lp2@gmail.com).

### **1ª Parte: Entendendo melhor a Linguagem Java**

Na primeira parte da aula, os professores conduzirão os alunos na execução de alguns passos de programação a fim de explorar: uso de package, saída de dados, compilação e execução, case-sensitive, tipos básicos, estruturas de desvio, um pouco de operadores em Java.

**Passo1:** Crie um programa em Java chamado IdadeParaVotacao para imprimir na saída padrão (monitor) uma frase indicando a sua idade e se você é um eleitor apto a exercer sua cidadania. Lembre-se de nomear seu arquivo corretamente, IdadeParaVotacao.java. Por enquanto, **não adicione package no seu programa**. (Você pode usar qualquer editor de texto de sua preferência)

**Passo2:** Agora vamos compilar e executar o programa IdadeParaVotacao.java.

Inicialmente, algumas dicas para trabalhar no Linux:

- Para criar um diretório use o comando `mkdir` e para passar de um diretório para outro use o comando `cd`. Veja exemplo:

```
$ mkdir nomeDoDiretorio
$ cd nomeDoDiretorio
nomeDoDiretorio$ mkdir ...
```

- Para ver o conteúdo de um diretório use o comando `ls`.

Abra um Terminal (ou console) no Linux, vá para o diretório onde o programa acima foi salvo (com o comando `cd`) e olhe o que tem dentro do diretório (com o comando `ls`). Apenas o código fonte (`IdadeParaVotacao.java`) estará lá. Agora tente compilá-lo usando o comando `javac`. Se seu programa não tiver erros de compilação, você vai conseguir compilá-lo e o arquivo de *bytecodes* (`IdadeParaVotacao.class`) vai ser gerado. Use novamente o comando `ls` para se certificar que o arquivo `IdadeParaVotacao.class` existe.

`IdadeParaVotacao.class` é o arquivo que contém os *bytecodes* do programa `IdadeParaVotacao.java`. Agora, a Java Virtual Machine (JVM) pode executar o seu programa interpretando blocos de comandos para a linguagem de máquina. Tente executar o seu programa, utilizando o comando: `java IdadeParaVotacao`. Lembre-se que pode ser necessário ensinar à Java Virtual Machine (JVM) em que diretórios ela deve procurar os seus programas (*classpath* ou `cp`, falamos sobre isso na aula de Programação 2, lembram?).

>> experimente mudar o nome da classe colocando todas as letras em minúsculo, sem modificar o nome do respectivo arquivo para perceber a consequência de Java ser case-sensitive.

**Passo3:** Vamos adicionar o programa `IdadeParaVotacao.java` no package `lp2.labGuiado01`. Vale lembrar que agora o programa tem um nome completo, que inicia com o nome do pacote e tem que estar numa árvore de diretórios idêntica ao que foi definido no pacote (`lp2/labGuiado01/IdadeParaVotacao.java`).

Crie a árvore de diretório `lp2/labGuiado01` a partir do diretório corrente. Mova o programa `IdadeParaVotacao.java` para o diretório `lp2/labGuiado01`. Vá para o diretório `lp2/labGuiado01` e compile novamente o programa.

Note que, a jvm deve executar o programa a partir do diretório pai do package. Assim para indicar o caminho a partir do qual a jvm vai procurar o pacote usando a opção `-cp` (`java -cp ../../lp2.labGuiado01.IdadeParaVotacao`).

>> experimente compilar e executar a partir de diretórios diferentes para exercitar mais o `-cp`

**Passo4:** Vamos modificar o programa `IdadeParaVotacao.java` para que ele calcule sua idade a partir do ano atual e do ano de nascimento. Nesse caso, você vai precisar criar uma variável para guardar o valor da idade que será resultado da diferença entre o ano atual e o ano de nascimento.

Java é uma linguagem fortemente tipada, assim, todas as variáveis precisam ter associado um tipo em tempo de compilação. Esta associação é feita da seguinte forma:

`<nome_do_tipo> <nome_da_variável>`

No caso da idade o tipo a ser usado será inteiro (`int`), mas Java suporta vários outros tipos primitivos como mostrado no material disponível [aqui](#).

Para calcular a idade será necessário usar o operador de subtração de Java (`-`). A linguagem também dispõe de outros operadores aritméticos, tais como: soma (`+`), multiplicação (`*`) e divisão (`/`). Ver mais informações [aqui](#).

Agora para mostrar o valor da variável que representa a idade será necessário usar o operador de concatenação de Java (`+`). Note que é o mesmo operador da soma aritmética, assim, este operador é dito "sobrecarregado", pois possui dois casos de uso. Então, a concatenação ocorre da seguinte forma:

```
System.out.println("texto 1" + variável);
```

Nesse caso, se um dos operandos for textual, o operador + fará concatenação; se os dois operandos forem numérico, o operador + fará soma aritmética.

Faça as modificações necessárias no programa, compile-o e execute-o novamente.

**Passo5:** Vamos modificar o programa IdadeParaVotacao.java para introduzir a checagem da idade e, assim, verificar se a pessoa está apta a votar ou não. Para tal, usaremos as estruturas de desvio de Java.

Um comando de desvio define uma condição em um programa, que permite que grupos de comandos sejam executados de maneira condicional, de acordo com o resultado da avaliação de um determinado teste (verdadeiro ou falso). Ou seja, programas utilizam comandos de desvio para escolher entre cursos alternativos de ações. A sintaxe do comando de seleção em Java é:

```
if (expressão) {  
    comando1;  
} else {  
    comando2;  
}
```

O comando1, será executado se o resultado da avaliação de expressão for verdadeiro (digamos  $x > 3$ ). Caso contrário (else) o comando2 será executado. Comentários:

- Vários comandos podem ser executados na cláusula if ou na cláusula else, bastando para isso que sejam usados os delimitadores de escopo { } (abre chaves e fecha chaves).
- Comandos de seleção podem existir dentro de outros comandos de seleção (na cláusula if ou else).

Faça as modificações necessárias no programa, compile-o e execute-o novamente.

**Passo6:** Observe que a mensagem apresentada pelo programa IdadeParaVotacao.java não contempla todos os casos de votação com exatidão, pois os jovens entre 16 e 18 anos têm o direito de votar, mas não a obrigação, enquanto que os maiores de 70 anos não tem mais obrigação de votar, embora possam continuar exercendo o seu o direito de votar. Modifique seu programa para tratar esses casos. Você vai precisar usar comandos de desvio aninhados.

Comandos de desvios são chamados aninhados quando eles são usados "dentro" de outros comandos de desvio. Em Java, podemos utilizar diferentes maneiras para escrever comandos aninhados. Abaixo, apresentamos a maneira mais tradicional e que permite um melhor entendimento do código com comandos de desvio aninhados.

```
if (condicao1) {  
    comando1;  
} else {  
    if (condicao2) {  
        comando2;  
    } else {  
        if (condicao3) {  
            comando3;  
        }  
    }  
}
```

Podemos utilizar também operadores lógicos para permitir que expressões mais complexas sejam avaliadas através de um conjunto de condições inter-relacionadas. Em Java podemos encontrar diversos operadores lógicos como pode ser visto [aqui](#).

## **2ª Parte: Programa Rápido**

Na segunda parte da aula, os alunos irão resolver um problema de programação individualmente.

**Questão.** Escreva um programa que calcule a idade de duas pessoas e informe a pessoa mais velha e a mais jovem. Considere o caso das pessoas terem idades iguais.