

Licenciatura em Engenharia Informática e de Telecomunicações
Departamento de Tecnologias de Informação e Comunicação

Programação I

Semana 2 (Aulas Prática 3 e 4)

Turmas: I11 e I15

Data: - Agosto- 2020

Docente: MSc, Rafael Beto Mpfumo

ESQUEMA AULA PRÁTICA

- Preparação do Ambiente *JAVA*:
Configuração de Variáveis de Ambiente;
Alguns comandos do *JAVA*.
Familiarização do *Notepad++*
- Contacto com o ambiente *Windows*: (*Shortcuts*; *Run*; Linha de Comandos)
- Resolução de exercícios.

NOTAS DA AULA TEÓRICA

- ✓ Um programa em *JAVA* é uma classe.
- ✓ Cada aplicação terá que possuir uma classe onde está definido o método **main** pelo qual se inicia a execução do programa.
- ✓ Por convenção o nome de uma classe começa por uma letra maiúscula.
- ✓ ; é um terminador de instruções.
- ✓ Os atributos e metodos das classes devem iniciar por minísculas e se estão formadas por várias palavras, se escrevem sem espaços e a primeira letra de cada palavra em maiúsculas.
- ✓ As constantes se escrevem em maiúsculas.
- ✓ { } delimitador de conjuntos de instruções.

ACTIVIDADE PRÁTICA 1 : Preparação do Ambiente *JAVA*

1. Baixar o *JDK*


Visite a [página de download do *JDK* no site da Oracle](#) e clique na imagem do *Java* ou no link do *JDK*, destacados em amarelo na imagem a seguir :

Oracle Technology Network / Java / Java SE / Downloads

[Java SE](#)
[Suporte para Java SE](#)
[Java Embedded](#)
[Java EE](#)
[Java ME](#)
[Java FX](#)
[Java DB](#)
[Web Tier](#)
[Comunidade](#)


[Resumo](#) | **[Downloads](#)** | [Documentação](#) | [Comunidade](#) | [Tecnologia](#) | [Formação](#)

Java SE Downloads



Java Platform (JDK) 8u111 / 8u112

[DOWNLOAD](#)



NetBeans com JDK 8

[DOWNLOAD](#)

Java Platform, Standard Edition

Java SE 8u111 / 8u112
 Java SE 8u111 includes important security fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release. Java SE 8u112 is a patch-set update, including all of 8u111 plus additional features (described in the release notes).
[Learn more](#)

Important planned change for MD5-signed JARs
 Starting with the April Critical Patch Update releases, planned for April 18 2017, all JRE versions will treat JARs signed with MD5 as unsigned. [Learn more and view testing instructions](#).
 For more information on cryptographic algorithm support, please check the [JRE and JDK Crypto Roadmap](#).

- [Installation Instructions](#)
- [Release Notes](#)
- [Oracle License](#)
- [Java SE Products](#)
- [Third Party Licenses](#)
- [Certified System Configurations](#)

Readme Files

- [JDK ReadMe](#)
- [JRE ReadMe](#)

JDK

[DOWNLOAD](#)

Server JRE

[DOWNLOAD](#)

JRE

[DOWNLOAD](#)

Which Java package do I need?

Java SDKs and Tools

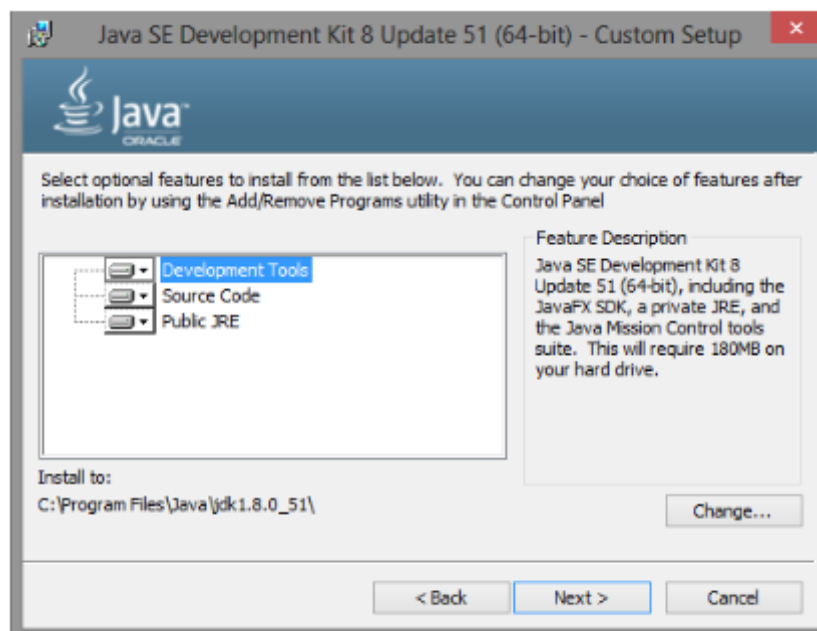
- [Java SE](#)
- [Java EE and Glassfish](#)
- [Java ME](#)
- [Java Card](#)
- [NetBeans IDE](#)
- [Java Mission Control](#)

Java Resources

- [Java APIs](#)
- [Technical Articles](#)
- [Demos and Videos](#)
- [Forums](#)
- [Java Magazine](#)
- [Developer Training](#)
- [Tutorials](#)
- [Java.com](#)

2. Instalação do *JDK*

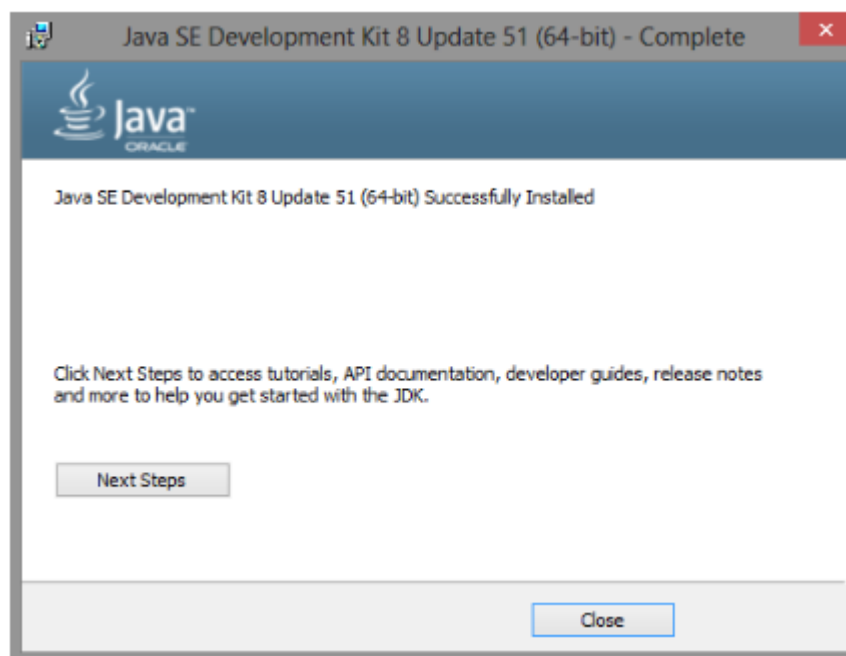
Após realizar o *download* do instalador do *JDK*, execute o mesmo e clique em *Next*:



O instalador irá então realizar algumas configurações e descompactar os arquivos necessários para a instalação. Quando concluído, a tela a seguir será exibida:



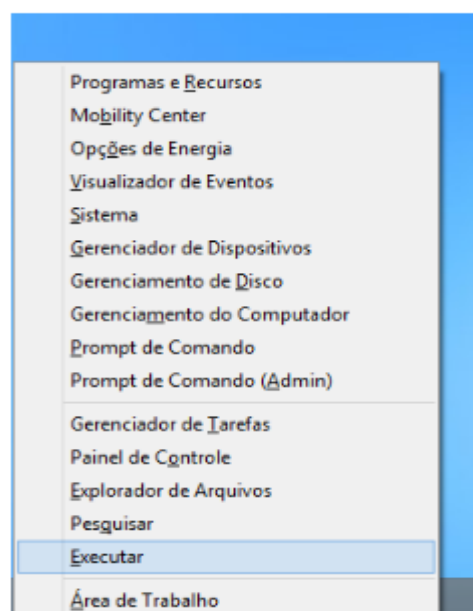
Quando a instalação estiver concluída, apenas clique em Close, conforme a imagem a seguir:



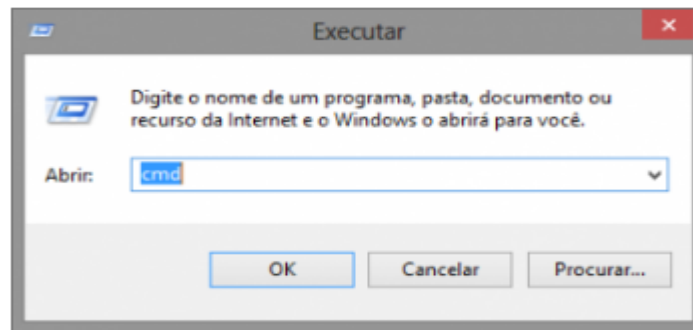
Para confirmar que após a instalação o JDK está disponível para o ambiente operacional, execute o prompt de comando do Windows (através do menu Executar ou através da opção Todos os Aplicativos do menu Iniciar) e execute o comando: `java -version`

Conforme as imagens a seguir:

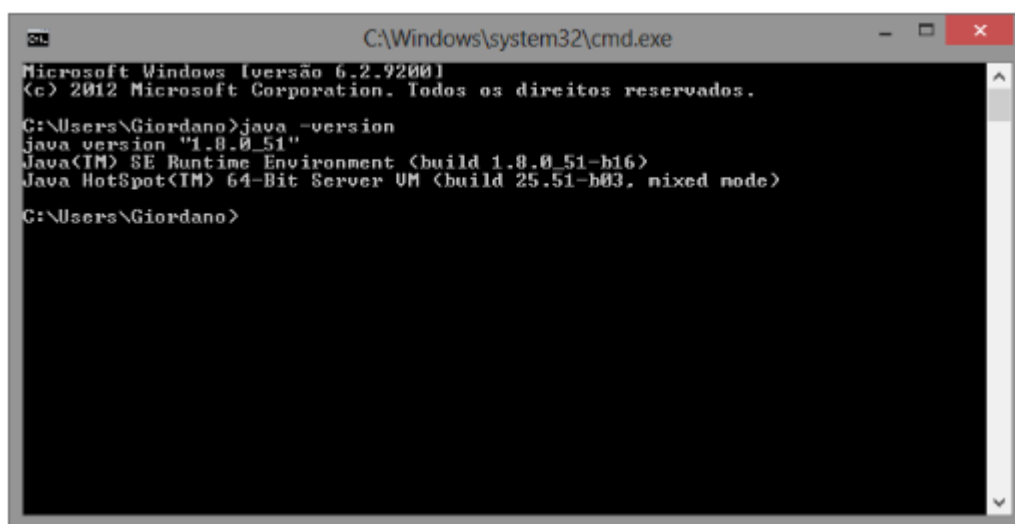
2.1 Menú de contexto do Windows, opção Executar (acessado através do atalho Windows + X):



2.2 Digitando o comando `cmd` na caixa de diálogo Executar:

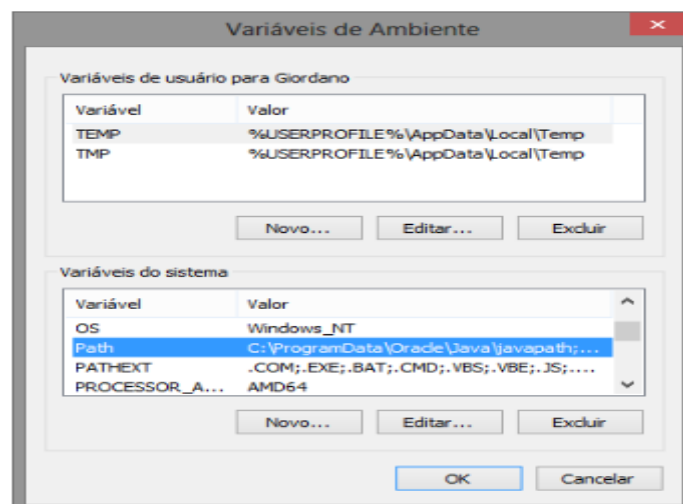


2.3 O *prompt* de comandos do *Windows* após a execução do comando *java -version*, exibe a versão actual do *java (JDK)* instalado:

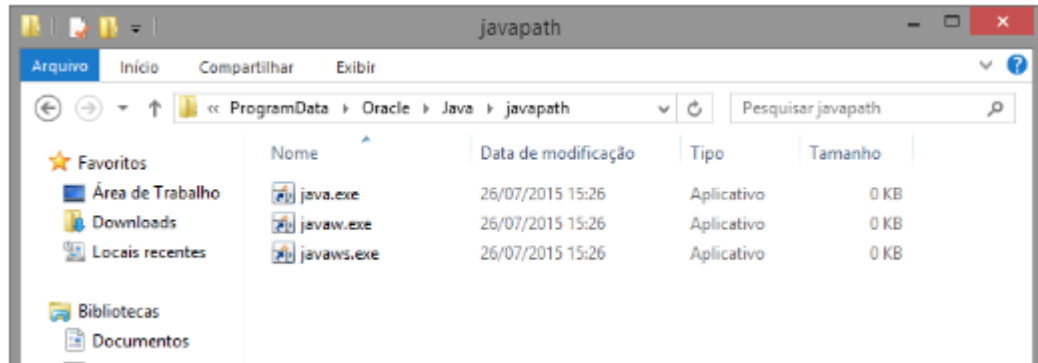


3. Configuração das variáveis de ambiente do *Windows*

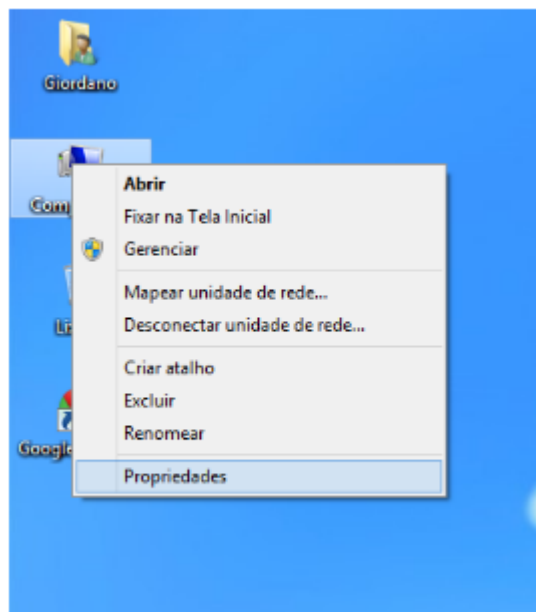
No *prompt* de comandos do *Windows*, só foi possível realizar a chamada ao compilador *java* por que o instalador do *java* cria uma entrada na variável ***Path*** do sistema, conforme a imagem a seguir:



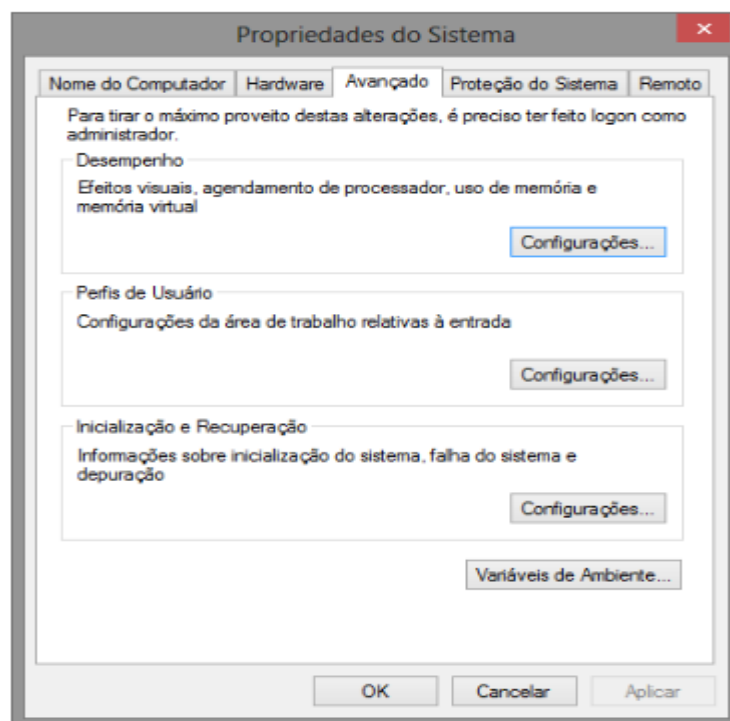
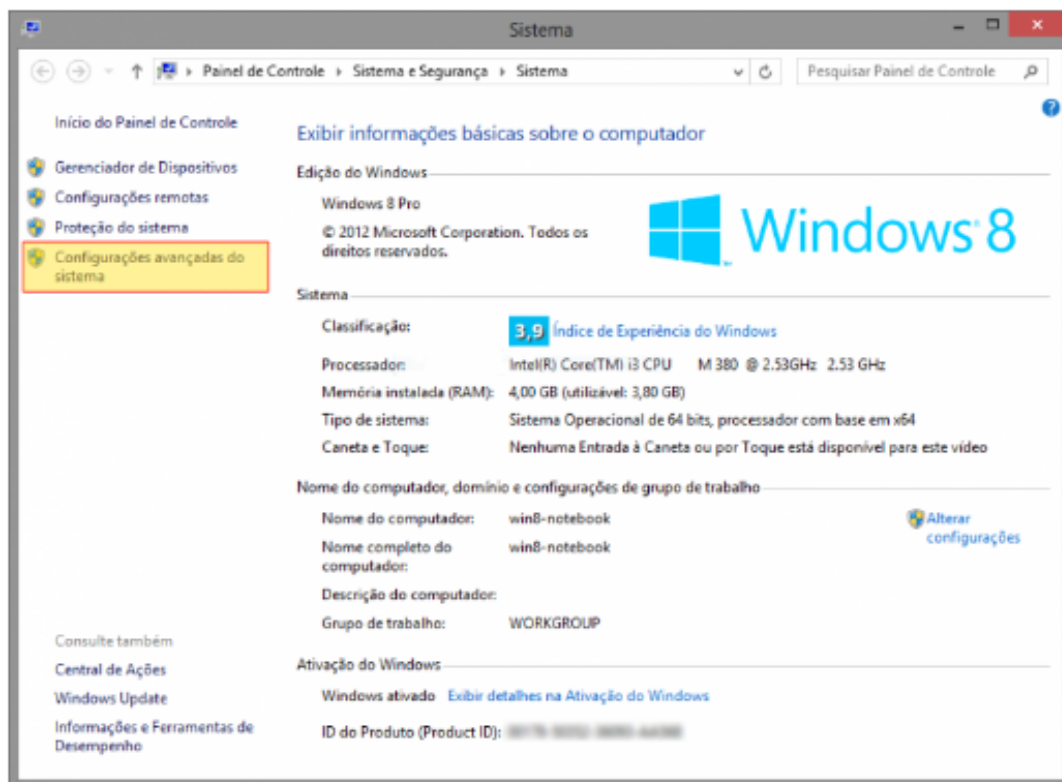
Se acessarmos este caminho que foi inserido no Path (No cado do Windows 8 – C:\ProgramData\Oracle\Java\javapath) veremos que a única coisa que ele fez foi criar atalhos para os executáveis do java:



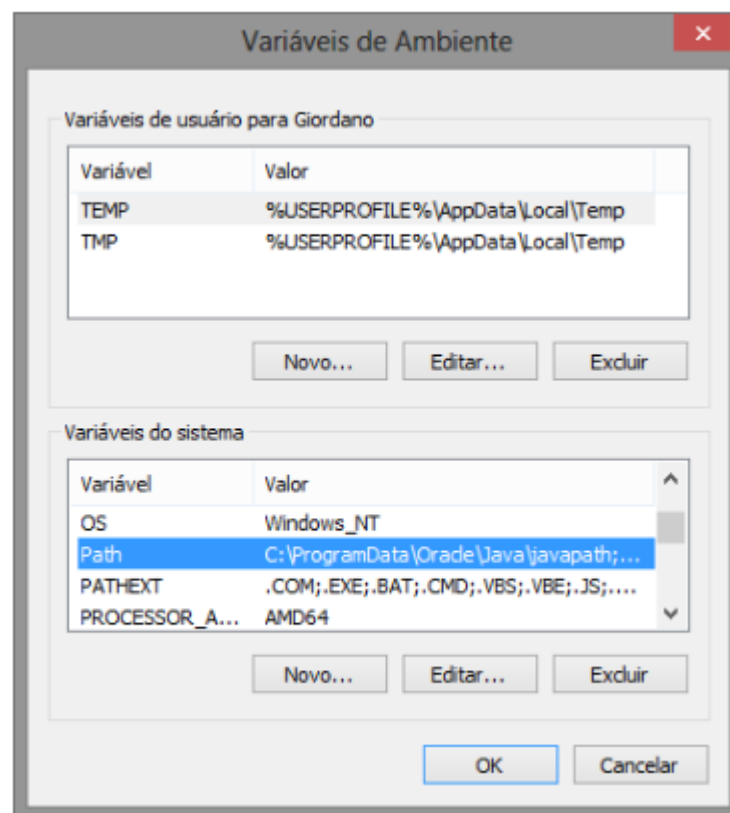
Mas geralmente as IDEs e ferramentas que utilizamos no processo de desenvolvimento de aplicativos Java requerem a existência de uma variável padrão chamada JAVA_HOME. Vamos fazer esta configuração então :).



Na janela que se abre, acesse a opção Configurações Avançadas do Windows, conforme destacado em amarelo:

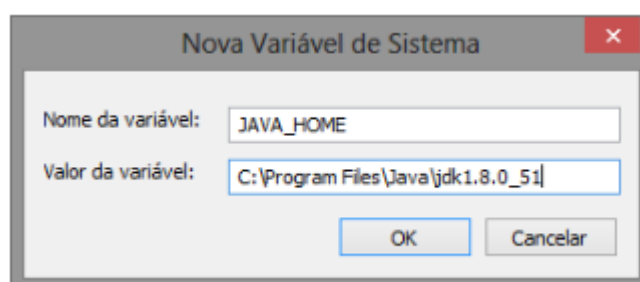


A janela de variáveis do ambiente será exibida conforme a seguir:

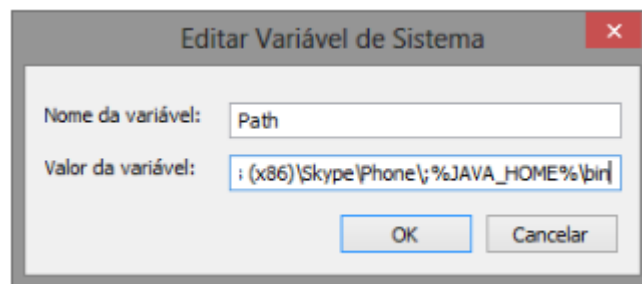


Nela, clique no botão Novo... na seção inferior (Variáveis do sistema). Na verdade este passo poderia ser executado na seção superior (Variáveis de usuário), mas por convenção, apliquemos esta configuração para todos os usuários do computador.

Na janela que se abre, definimos a variável `JAVA_HOME`, conforme a imagem abaixo, informando no valor da variável, aquele caminho de instalação do JDK que anotamos no início do processo de instalação:



Agora selecionamos a variável Path na lista de variáveis do sistema e clicamos no botão Editar... Tome cuidado para não apagar nada do que já está escrito no valor da variável Path, senão diversas aplicações que dependem destas configurações podem parar de funcionar corretamente. O que deve ser feito no valor desta variável é ir até o final do valor da mesma e acrescentar a nossa variável `JAVA_HOME` ao Path, conforme a imagem a seguir:



Observe que o texto adicionado no fim do valor foi ;%JAVA_HOME%\bin
Após a conclusão dessas etapas, seu ambiente de desenvolvimento Java está configurado correctamente.

PARTE A

1. Qual o fundamento da Linguagem de Programação *Java*.
2. Com relação a linguagem *Java* indique que componentes executam as seguintes funções :
 - a) compilação, depuração e execução um programa.
 - b) converter bytecode em código específico de máquina.
 - c) é responsável por executar o programa java.
3. O que são *bytecodes* ?
4. Que função desempenha o *JDK Tools*.
5. O que representam as variáveis do ambiente em *Java*.
6. Qual é a estrutura de um programa em *Java*.
7. Para que servem os comandos *java* e *javac* ?
8. Em que consiste o processo de configuração das variáveis do ambiente em *Java*. Descreva este procedimento.
9. O que representa uma variável ?
10. Quais são os tipos de dados primitivos.
11. Qual é a diferença existente entre operadores unários e binários ?
12. Assinale com C os identificadores correctos e com I os incorrectos. Explique o que está errado nos identificadores incorrectos.
 - () valor
 - () _b248

- () nota*do*aluno
- () a1b2c3
- () 3 x 4
- () Maria
- () km/h
- () xyz
- () nome empresa
- () sala_215
- () "nota"
- () ah!
- () 1AB
- () A?

13. Escreva o tipo de dado ideal para se representar cada uma das seguintes informações:

- (a) O nome de uma rua
- (b) A data de nascimento de uma pessoa
- (c) Se uma pessoa é diabética ou não
- (d) O saldo de uma conta bancária
- (e) O resultado de uma operação de raiz quadrada

14. Identifique o tipo de dados dos seguintes valores:

- (a) "9 de agosto de 1968"
- (b) 1.3
- (c) falso
- (d) -31
- (e) "?"

PARTE B

1. Como forma de ambientação com a linguagem *JAVA*, comecemos por representar uma classe, (como ilustra o programa abaixo) com o nome de **Primeiro**, usando o editor do *NotePad* para depois compilar e executar o mesmo usando a linha de comando (*CMD*), de forma a analisar o processo:

```
Public class Primeiro {  
    public static void main(String [] args) {  
        System.out.println("Este é o 1º programa em Java");  
//imprimir na ecrã  
    }  
}
```

2. Desenvolva um programa em *Java* que imprima na tela a seguinte tabela de equivalência das medidas de armazenamento de dados em computadores :

Medida	Simbologia	Equivalencia	Equivalencia en Bytes
byte	b	8 bits	1 byte
kilobyte	Kb	1024 bytes	1024 bytes
megabyte	MB	1024 KB	1 048 576 bytes
gigabyte	GB	1024 MB	1 073 741 824 bytes
terabyte	TB	1024 GB	1 099 511 627 778 bytes
petabyte	PB	1024 TB	1 125 899 906 842 624 bytes
exabyte	EB	1024 PB	1 152 921 504 606 846 976 bytes
zetabyte	ZB	1024 EB	1 180 591 620 717 411 303 424 bytes
yottabyte	YB	1024 ZB	1 280 925 819 614 629 174 706 176 bytes
brontobyte	BB	1024 YB	1 237 940 039 285 380 274 899 124 224 bytes
geopbyte	GB	1024 BB	1 267 650 600 228 229 401 496 703 205 376 bytes

3. Desenvolva um programa em que mostre o seu horário de aulas.
4. Escreva um programa em *Java* que permita calcular o volume de um cone, conhecidos o seu raio a sua altura.
5. Desenvolva um programa em *Java* que permita criar uma variavel nome, atribuindo o seu nome completo, e duas variaveis x e y com os valores 2562 e 1415 respectivamente. O programa deve efectur a soma, multiplicação, subtração, divisão e resto da divisão entre dois números inteiros fornecidos e mostrar na tela o seu nome com os resultados obtidos.
6. É necessário determinar o salário de um trabalhador com base nas horas que ele trabalhas, no pagamento por hora que ele aufer, nos descontos salarias que ele recebe (11% de IRPS e 2% de INSS). Represente um programa em *Java* para representar a solução correspondente.
7. Crie um programa em *java* que permita introduzir a kilometragem percorrida por um determinado automóvel e valor de combustível, sabendo que o litro de gasolina custa 61mtn. Calcule o gasto sabendo que o veiculo percorre 8Km/l.
8. Implemente um programa em *Java* para determinar aproximadamente quantos meses, semanas, dias e horas um individuo viveu, conhecida a sua data de nascimento.
9. É necessário encher um tanque com capacidade de 50 metros cúbicos. Represente um programa em *Java* que calcula as horas necessárias para encher o tanque com uma mangueira que tenha uma capacidade de L litros de água por minuto.
10. O custo de um carro importado do Japão, para um comprador final é a soma total do CIF (custo do veículo no parque de origem, custo de seguros, custo do frete de transporte) adicionado aos Impostos (serviços de importação, direitos

aduaneiros), despesas bancárias estipulada pelo banco, para o despachante aduaneiro, transpasse (manuseamento e entrega da viatura no porto) do carro pela Maputo Car Terminal Lda., scanner do carro pela KUDUMBA, serviços do INATTER (emissão de chapa matricula e livrete) e a porcentagem do lucro do vendedor (sobre o preço de venda). Assuma o lucro de um vendedor de 7% em cada unidade vendida. Represente um programa em Java que permita calcular o preço final que um cliente podera pagar um carro, e o lucro do venedor por venda.