



Nota: 10.0



Tempo: Terminou



Sair

Laboratório 3: Introdução à Orientação a Objetos

Objetivo

- Modelagem, implementação, compilação e execução de **classes** em java usando o Eclipse.

Descrição

- Neste trabalho, você implementará as **classes** necessárias para representar um *computador*.
- Irá descrever **atributos** e implementar **métodos** e **construtores** com base em um **diagrama de classes** da UML.
- Em seguida, **objetos** serão criados a partir das **classes** implementadas.
- Nesta **abstração**, um computador é constituído de *processador*, *memória* e *disco*.

Passos Iniciais

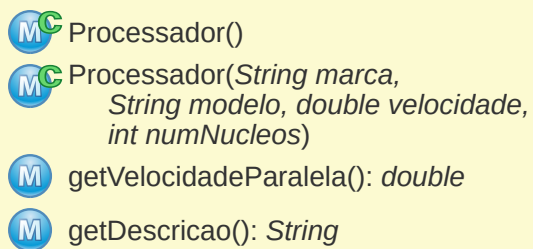
- Inicie o Eclipse. Vá em "File" → "New" → "New Java Project". Nome do projeto: "Lab-IntroducaoP00".
 - Na parte *Module* (parte de baixo da janela), desselecione "Create module-info.java file".
- Nas questões a seguir, será pedido para criar diversas classes. Para isso, vá em "File" → "New" → "Class".
 - Deixe o campo "**package**" em branco.
- O Eclipse compila automaticamente as classes sempre que você salva. Para executar e testar, basta ir em "Run" → "Run" (Ctrl+F11).
- A saída e entrada do programa será feita através da aba "**Console**" do Eclipse.

Questão 1: Classe Processador



Processador

marca: *String*modelo: *String*velocidade: *double*numNucleos: *int*



Crie uma classe para representar um **Processador** de acordo com o diagrama de classe ao lado. Os atributos de um objeto da classe **Processador** poderão ser setados diretamente (e.g., `proc1.marca="Intel"`) ou pelo *método construtor*. Use a técnica de *encadeamento de construtores* para criar os dois construtores da classe, semelhante ao feito em sala e mostrado nos slides. Implemente os métodos:

- `getVelocidadeParalela`: retorna um **double** contendo a multiplicação da velocidade do processador pela quantidade total de núcleos.
- `getDescricao`: retorna uma **String** contendo a descrição do processador de acordo com a linha a seguir:

```
Processador: marca=Intel, modelo=i7, velocidade=3.2GHz, numNucleos=8, velocidadeParalela=25.6GHz.
```

Para testar a classe, crie uma nova classe chamada **ComputadorMain**. Nesta classe, crie o método `main`, que será o ponto de partida do seu programa. No método `main`, crie um ou mais objetos da classe **Processador** e, em seguida, imprima o resultado da execução do método `getDescricao` dos objetos criados.

Para essa questão, submeta apenas a classe **Processador**. Não precisa submeter a classe **ComputadorMain**.

Dica:

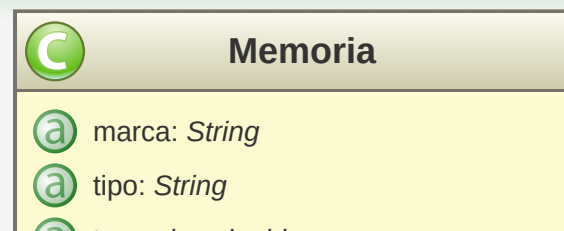
- Um *número real* em Java (e.g., `"8.4"`) é, por padrão, do tipo **double**. Para escrever um número que seja do tipo **double**, deve-se usar um `"f"` no final (e.g., `"8.4f"`).




O prazo de entrega do trabalho terminou. Portanto, o botão abaixo está desabilitado.








Enviar "Processador.java"

Solução correta!

Questão 2: Classe Memória



 tamanho: *double*
 velocidade: *double*
 numPentes: *int*

  Memoria()
  Memoria(*String* marca, *String* tipo,
double tamanho,
double velocidade, *int* numPentes)
 getTamanhoTotal(): *double*
 getVelocidadeParalela(): *double*
 getDescricao(): *String*

De forma semelhante à questão anterior, implemente uma classe para representar a **Memoria** do computador de acordo com o diagrama de classe ao lado. Implemente os métodos:

- getTamanhoTotal: retorna um **double** contendo a multiplicação da quantidade pelo tamanho dos pentes de memória.
- getVelocidadeParalela: retorna um **double** contendo a multiplicação da quantidade pela velocidade dos pentes de memória.
- getDescricao: retorna uma **String** contendo a descrição da memória de acordo com a linha a seguir:

```
Memoria: marca=Kingston, tipo=DDR4, tamanho=8.0GB, velocidade=3.2GHz, numPentes=4, tamanhoTotal=32.0GB,  
velocidadeParalela=12.8GHz.
```

Para testar esta segunda classe, modifique a classe **ComputadorMain**, criada na questão anterior para criar um ou mais objetos da classe **Memoria** e, em seguida, imprimir o resultado da execução do método getDescricao dos objetos criados.

Para essa questão, submeta apenas a classe **Memoria**. Não precisa submeter a classe **ComputadorMain**.

Dica:

- Note que não estamos usando acentos nas strings.

O prazo de entrega do trabalho terminou. Portanto, o botão abaixo está desabilitado.

Enviar "Memoria.java"

Solução correta!

Questão 3: Classe Disco





Disco



marca: *String*



tipo: *String*



capacidade: *double*



rpm: *int*



Disco()