

Sua nota: **81,81%**A última nota: **81,81%** • A nota mais alta: **81,81%** • O mínimo para ser aprovado é 75%. Sua maior nota será mantida.[Próximo item →](#)

1. É um método especial de uma classe Java que tem o mesmo nome da classe, cuja execução se dá imediatamente após a instanciamento de um objeto da classe, com o objetivo de alocar memória e iniciar as suas variáveis de instância. Como é chamado esse método especial? **[Na sua resposta, por limitação do presente ambiente de correção automática, coloque obrigatoriamente apenas a primeira letra em maiúscula; o restante deverá estar em minúscula!]**

1 / 1 ponto

Construtor

Correto

2. Uma classe que tem um método abstrato não precisa ser declarada como abstrata. Verdadeiro/V ou Falso/F?

1 / 1 ponto

F

Correto

Uma classe que tem um método abstrato precisa ser declarada como abstrata.

3. Uma classe abstrata pode não ter nenhum método abstrato.

1 / 1 ponto

☐ Falso☒ Verdadeiro

Correto

Basta que a classe seja declarada como abstrata; nesse caso, o desenvolvedor usualmente deseja transmitir para as subclasses uma interface comum mínima.

4. O código abaixo compilará corretamente **(indique apenas e exatamente V ou Verdadeiro ou Sim - F ou Falso ou Não; e não deve justificar)**. **Obs.: se você escrever as iniciais da sua resposta em minúsculas (v ou verdadeiro ou sim - f ou falso ou não), o sistema não entenderá e marcará como errado, mesmo se for a resposta certa!** [Supondo que o trecho representado por "... " compila corretamente]

1 / 1 ponto

```
1 public abstract class X {...}
2 public class Teste{
3     public static void main (String[] args) {
4         X x = new X();
5     }
6 }
```

F

Correto

Classes abstratas não podem ser instanciadas, de modo que "new X()" não compilará!

5. O código abaixo compilará corretamente **[supondo que o trecho representado por "... " compila corretamente]**:

1 / 1 ponto

```
1 class Teste{
2     public void metodo (int i) {...}
3     protected void metodo (double x) {...}
4 }
```

☐ Falso☒ Verdadeiro

Correto

Nesse caso, o overloading ou sobrecarga é legítimo, pois os argumentos dos dois métodos com mesmo nome são de tipos diferentes!

6. O código abaixo compilará corretamente **(indique apenas e exatamente V ou Verdadeiro ou Sim - F ou Falso ou Não; e não deve justificar)**. **Obs.: se você escrever as iniciais da sua resposta em minúsculas (v ou verdadeiro ou sim - f ou falso ou não), o sistema não entenderá e marcará como errado, mesmo se for a resposta certa!** [Supondo que o trecho representado por "... " compila corretamente]

1 / 1 ponto

```
1 class Teste{
2     public int metodo ( ) {...}
3     protected double metodo ( ) {...}
4 }
```

F

Correto

Nesse caso, o overloading ou sobrecarga não é legítimo, pois os dois métodos com mesmo nome não possuem argumentos! Dito de outra forma, o número de argumentos é nulo e igual em ambos os

métodos.

7. É um mecanismo existente

1 / 1 ponto

no paradigma orientado a objetos que permite a reutilização da estrutura e do comportamento de uma classe ao se definir novas classes; é conhecido também como relacionamento "é um"; a classe que herda o comportamento é chamada de subclasse e a que definiu o comportamento, superclasse. Qual é o nome desse mecanismo?

- ☐ Método
- ☐ Responsabilidade
- ☐ Polimorfismo
- ☒ Herança

✔ Correto

Embora a maioria das linguagens orientadas a objetos atuais permitam que se faça, não é boa prática permitir que a estrutura de uma classe (as variáveis de instância) sejam herdadas por subclasse, uma vez que por convenção ela deve ser declarada como privada na classe! Além disso, apenas comportamento público ou protegido é herdado por subclasse!

8. Apresente como é em Java a assinatura do método correspondente à primeira mensagem que aparece no trecho do exemplo abaixo **[se você vislumbrar mais de uma assinatura possível, apresente apenas uma delas!]** [A assinatura de método em Java tem um formato que não segue a sintaxe do Java; a sintaxe é a de assinatura de método, que poderá ser usada ao se avaliar programas em C# ou Python, por exemplo!]:

0 / 1 ponto

```
1  . . .
2  Ponto ponto1 = new Ponto( );
3  Ponto ponto3 = new Ponto( );
4  . . .
5  if (ponto1.igual(ponto3))
6  |   ponto3.mover(5,10);
7  . . .
8
```

C#

✘ Incorreto

9. Os dados e operações de um

1 / 1 ponto

objeto que são visíveis externamente compreendem seu/sua _____.

- ☒ Interface
- ☐ Sobreposição
- ☐ Método
- ☐ Sobrecarga

✔ Correto

10. O princípio de _____ permite que

1 / 1 ponto

objetos que pertencem a diferentes classes respondam de forma distinta a mensagens idênticas. **[Na sua resposta, por limitação do presente ambiente de correção automática, coloque obrigatoriamente apenas a primeira letra em maiúscula; o restante deverá estar em minúscula!]**

Polimorfismo

✔ Correto

11. A principal diretriz convencionada

0 / 1 ponto

para atributos de objetos é que todas as variáveis de instância devem ser declaradas _____.

- ☐ Públicas
- ☒ Protegidas
- ☐ Privadas

✘ Incorreto