

AP03 (tipo teste) Relacionamentos 2

Entrega 9 out em 23:59**Pontos** 10**Perguntas** 10**Disponível** 27 set em 0:00 - 9 out em 23:59**Limite de tempo** Nenhum**Tentativas permitidas** Sem limite

Instruções

Atividade prática tipo teste - Relacionamentos 2 (mais um pouco de associações, dependências e herança)

importante: atividade pontuada com peso 2,0 na avaliação APs que compõe a nota final

[Fazer o teste novamente](#)

Histórico de tentativas

	Tentativa	Tempo	Pontuação
MAIS RECENTE	Tentativa 1	13 minutos	9 de 10

Pontuação desta tentativa: **9** de 10

Enviado 4 out em 21:12

Esta tentativa levou 13 minutos.

Pergunta 1

0 / 1 pts

Considere o trecho de código em java abaixo, referente a implementação de uma classe "A.java":

```
1 public class A {  
2  
3     C c;  
4  
5     void metodo(B b) {  
6  
7         D d = new D();  
8  
9     }  
10 }
```

Neste trecho é possível identificar o relacionamento da classe A com outras classes do projeto (B, C e D). Conforme apresentado pelo código podemos dizer que os relacionamentos entre A e as demais classes (B, C e D) são respectivamente:

☐ associação, uso e dependência

☐ dependência, uso e associação

☐ associação, dependência e uso

Você respondeu

☒ uso, dependência e associação

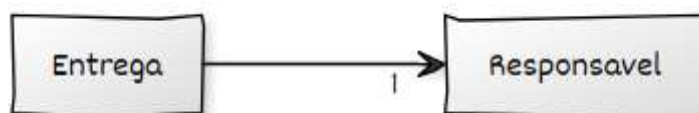
Resposta correta

☐ dependência, associação e uso

Pergunta 2

1 / 1 pts

Considere a imagem de um relacionamento entre duas entidades abaixo:



é uma **implementação válida** para receber (especificamente) esse relacionamento:

```
public class Entrega {  
    ArrayList<String> responsaveis = ArrayList<String>();  
  
    }  
}
```

```
public class Responsavel {  
    Entrega entrega;  
  
    }  
}
```

Correto!

```
public class Entrega {  
    Responsavel responsavel;  
  
    }  
}
```

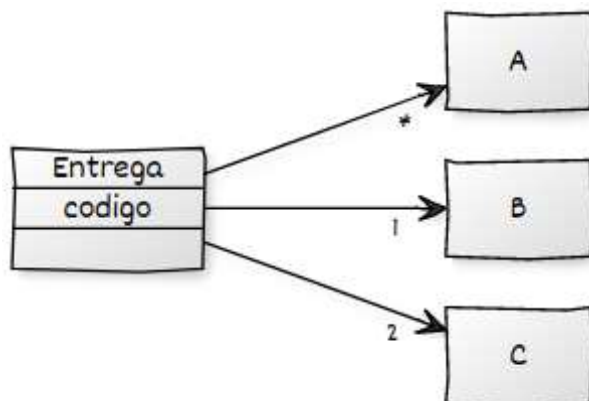
```
public class Responsavel {  
    ArrayList<Entrega> entregas = ArrayList<Entrega>();  
  
    }  
}
```

```
public class Entrega {  
    ArrayList<Responsavel> responsaveis = ArrayList<Responsavel>();  
  
    }  
}
```

Pergunta 3

1 / 1 pts

Considere a imagem UML abaixo:



sabendo que a implementação do protótipo da classe Entrega esta representado conforme segue:

```
1 public class Entrega {  
2  
3     String codigo;  
4     Responsavel responsavel;  
5     Endereco[] enderecos = new Endereco[2];  
6     ArrayList<Produto> produtos = ArrayList<Produto>();  
7  
8 }
```

podemos assumir que A, B e C, são respectivamente:

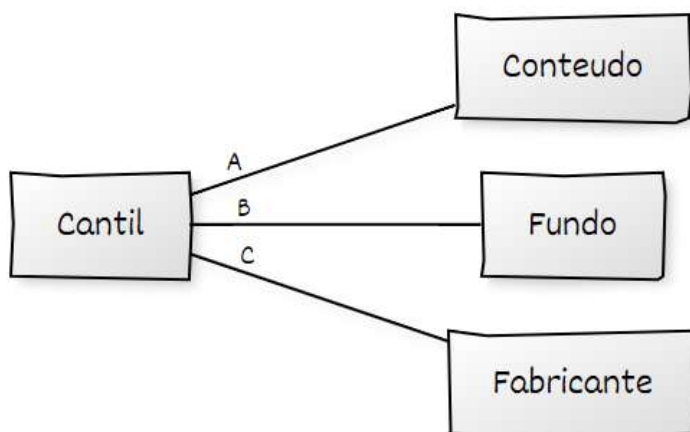
Correto!

- ☒ Produto, Responsavel e Endereco
- ☐ Endereco, Produto e Responsavel
- ☐ Responsavel, Produto e Endereco
- ☐ Produto, Endereco e Responsavel
- ☐ Endereco, Responsavel e Produto

Pergunta 4

1 / 1 pts

Considere a estrutura de classe abaixo.



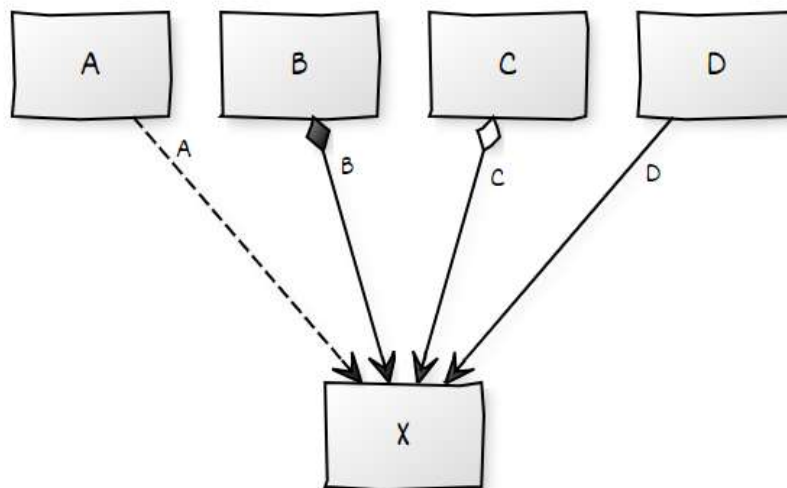
Assumindo que neste contexto (acima) temos, necessariamente, uma Associação, uma Agregação e uma Composição entre A, B e C; respectivamente, A, B e C é melhor representado por:

Correto!

- ☐ associação, agregação e composição
- ☐ associação, composição e agregação
- ☐ agregação, associação e composição
- ☒ agregação, composição e associação
- ☐ composição, associação e agregação

Pergunta 5**1 / 1 pts**

Considere a estrutura de classe abaixo.



Segundo sintaxe da UML para diagramas de classe, A, B, C e D são respectivamente:

Correto!

- ☐ associação, composição, agregação e dependência
- ☒ dependência, composição, agregação e associação
- ☐ composição, dependência, agregação e associação
- ☐ dependência, agregação, composição e associação
- ☐ dependência, composição, associação e agregação

Pergunta 6**1 / 1 pts**

Associe os seguintes **fundamentos** OO

- **(A) Abstração**
- **(B) Dependência**

às suas **definições** abaixo:

1. **uma princípio de análise de projeto**
2. **um relacionamento entre classes**

desta forma temos, escolha uma alternativa:

Correto!☒ A1 e B2☐ A2 e B1**Pergunta 7****1 / 1 pts**

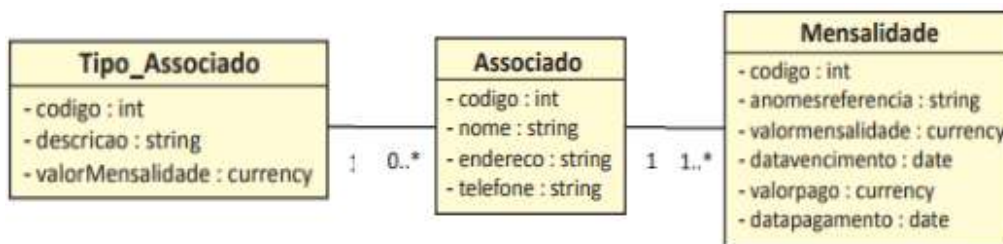
Em um Diagrama de Classe da UML, **NÃO** é possível especificar:

1. relacionamento entre classes;
2. atributos privados de classe;
3. código e lógica de implementação;

Correto!☒ 3☐ 2☐ 1

Pergunta 8**1 / 1 pts**

No desenvolvimento de um sistema de automação para um clube recreativo, serão implementados módulos para o departamento administrativo, tais como os módulos de cadastro de associados e de gestão de mensalidades. Na construção desses módulos, algumas regras de negócio serão aplicadas conforme o diagrama abaixo.



Com base nesse diagrama, avalie as afirmações a seguir.

- (I) Haverá somente 12 mensalidades para cada associado, ou seja, as mensalidades do associado não poderão ultrapassar os 12 meses do ano.
- (II) Não existem associados sem mensalidades, nem mensalidades sem associado a elas relacionado.
- (III) Poderão existir tipos de associados que não terão nenhum associado relacionado a eles.

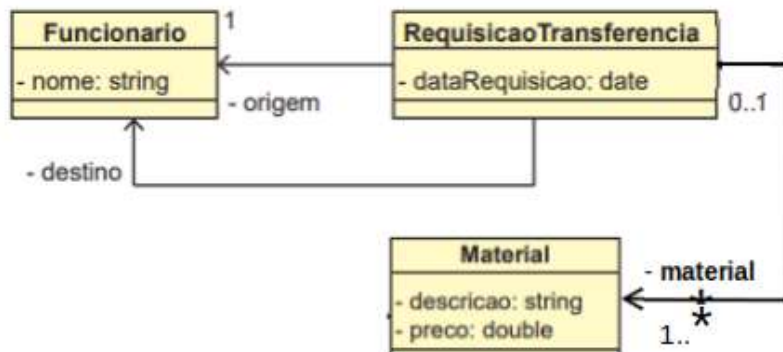
É correto apenas o que se afirma em:

☐ I e III

☐ I, II e III

Correto!☐ I e II☒ II e III☐ Nenhuma esta correta**Pergunta 9****1 / 1 pts**

Considere o diagrama de classe abaixo:



Proponha a implementação de um construtor para a classe **Funcionario** segundo as opções abaixo.

Correto!

☒ `public Funcionario(String param) {
 nome = param;
}`

☐ `private int Funcionario() {
 String param;
}`

☐ `public Funcionario(String departamento) {
 this.departamento = departamento;
}`

☐ `public void Funcionario(int param) {
 nome = param;
}`

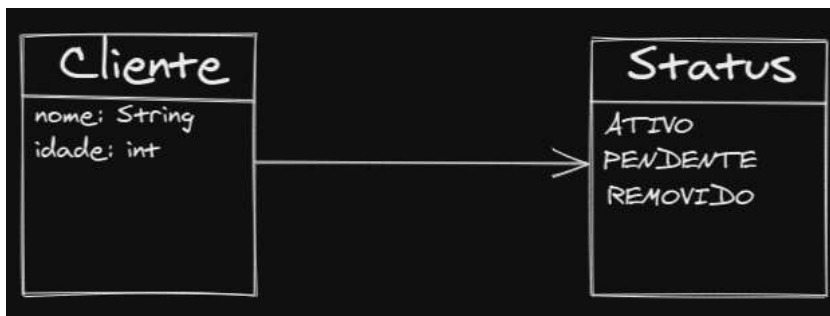
☐ `public void Constutor() {
 String nome;
}`

Pergunta 10

1 / 1 pts

Considere a representação UML abaixo,

que representa duas classes de negócio, seus atributos e o relacionamento existente entre elas:



o **entendimento** sobre o relacionamento representado é que Cliente...

Correto!

☒ TEM Status

☐ EXECUTA Status

☐ USA Status

Pontuação do teste: **9** de 10