

Atskaitančiojo kodas (vidko) 1 3 5 7 9

(pildo studentas)

Studijų modulio pavadinimas, kodas

T120B029 Programų sistemų analizės ir projektavimo įrankiai

Atsakingo dėstytojo vardas, pavardė

Eduardas Bareša

Egzamino data

(pildo atsakings dėstytojas)

2020-06-18

Egzamino pradžia

(pildo atsakings dėstytojas)

9:00

Egzamino trukmė

(pildo atsakings dėstytojas)

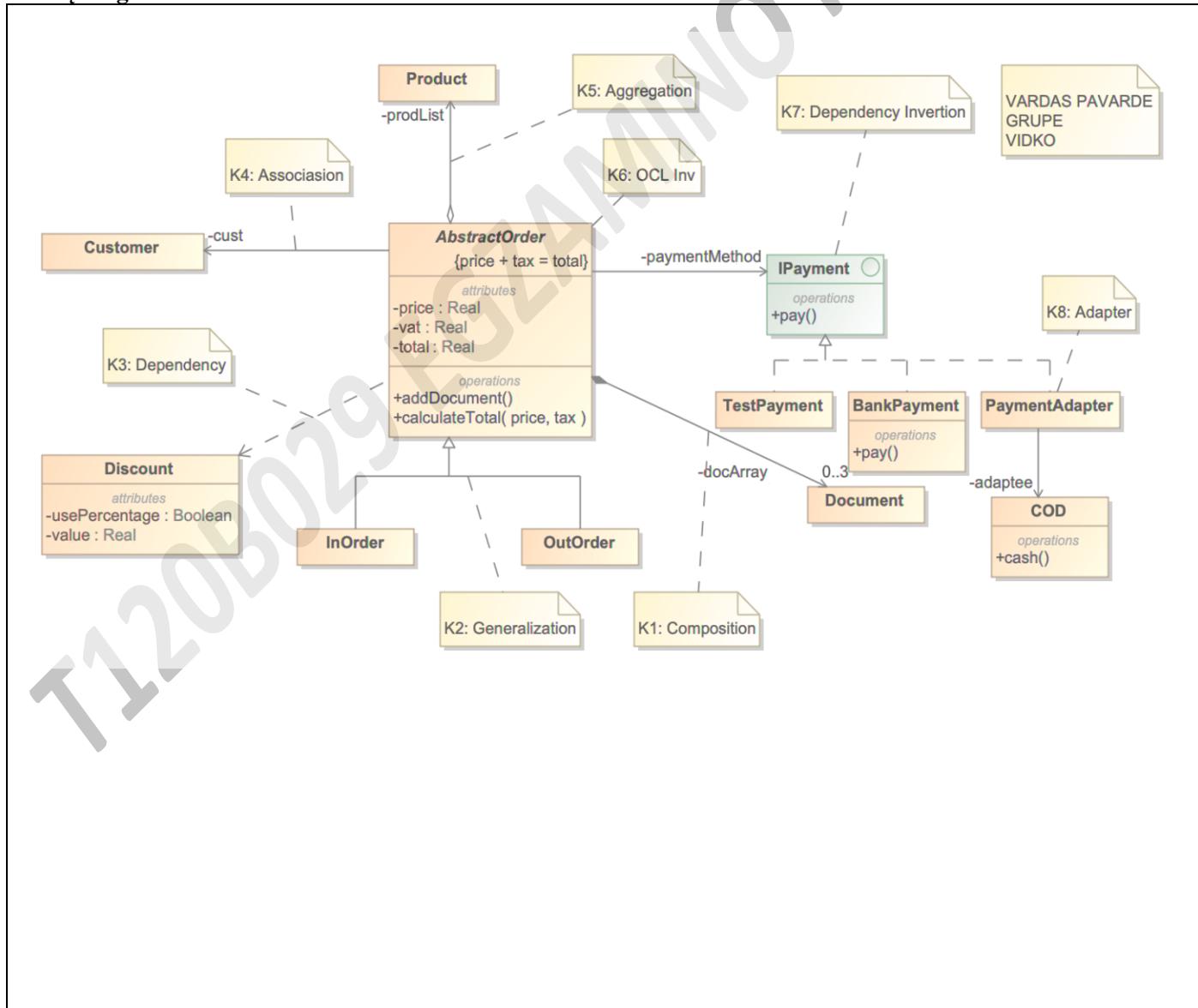
1,5 val.

Instrukcija atskaitančiam:

- Diagramoje pastaboje išrašykite vardą, pavardę, grupę ir VIDKO.
- Naudodami MagicDraw, atsakant į žemiau pateiktus klausimus, sujunkite klasų diagramoje klases bei sąsajas atitinkamais ryšiais. Jei reikia, papildykite diagramą klasėmis ir/arba sasažomis.
- Ryšius diagramoje pažymėkite pastabomis (note) su klausimų numeriais.
- Išelkite papildytą klasų diagramą į atsakymų failą.
- Naudodami teksto redaktorių, parašykite pseudokodą, kuris atspindėtų klasų bei ryšių esmę ir metodų realizaciją. Išelkite pseudokodą į atsakymų failą.
- Atsakymų failą užsaugokite PDF formatu, failo pavadinime nurodant vardą ir pavardę.

Dalykinė sritis: Pirkimo ir pardavimo užsakymai

Klasų diagrama:



#	Klausimas	Balai
1	Susiekite užsakymo (AbstractOrder) ir Dokumento (Document) klasses kompozicijos ryšiu, nustatančiu trijų dokumentų rinkinį prie užsakymo, nurodykite kardinalumus. Pridėkite metodą addDocument().	1

Pseudokodas

```
class abstract AbstractOrder{
    Private Document docArray[3];
    Public void addDocument(Document newDoc) {
        If(docArray.Size < 3) {
            docArray[docArray.Size] = newDoc;
        }
    }
}
```

#	Klausimas	Balai
2	Susiekite užsakymo (AbstractOrder) abstrakčią klasę ir konkretias užsakymų klasses (InOrder, OutOrder) paveldėjimo „generalization“ ryšiais.	1

Pseudokodas

```
class InOrder extends AbstractOrder {
}

Class OutOrder extends AbstractOrder {
```

#	Klausimas	Balai
3	Susiekite užsakymo (AbstractOrder) ir nuolaidos (Discount) klasės priklausomybės „dependency“ ryšiu, pridėkite naują metodą <code>ApplyDiscount()</code> su atitinkamais parametrais.	1

Pseudokodas

```

class abstract AbstractOrder{

    public void ApplyDiscount(Discount d) {
        if(d.usePercentage) {
            price = price(100 - d.value)/100;
        }else{
            price = price - d.value;
        }
    }
}

```

#	Klausimas	Balai
4	Susiekite užsakymo (AbstractOrder) ir užsakovo (Customer) klasės asociacijos ryšiu.	1

Pseudokodas

```

class abstract AbstractOrder {
    Private Customer cust;
}

Class Customer{
}

```

#	Klausimas	Balai
5	Susiekite užsakymo (AbstractOrder) ir prekės (Product) klasses agregacijos ryšiu.	1
Pseudokodas		
<pre>class abstract AbstractOrder { Private List<Product> prodList; }</pre>		

#	Klausimas	Balai
6	Parašykite OCL invariantą, kad bendra užsakymo sumą (total) lygi kainos (price) ir mokesčių (tax) sumai. Parašykite vienetų (unit) testo pseudokodą šiam apribojimui patikrinti.	1,5
OCL kodas		
<pre>context AbstractOrder inv validateTotal: price + tax = total</pre>		
Unit Test pseudokodas		
<pre>public class AbstractOrderTests { [Test] public void TestTotal() { int price = 100; int tax = 21; int total = 121; AbstractOrder order = new InOrder(); calculator.PressEquals(); Assert.AreEqual(total, Order.calculateTotal(price, tax)); } }</pre>		

#	Klausimas	Balai
7	Papildykite diagramą, pritaikant „dependency inversion“ principą, realizuotą per metodą addPayment (), taip, kad užsakymas (AbstractOrder) nepriklausytu nuo bankinio mokėjimo metodo (BankPayment). Pritaikykite testavimo valdiklį.	2

Pseudokodas

```

Interface IPayment{
    Public bool pay();
}
Public abstract class AbstractOrder {
    Private IPayment paymentMethod;
    Public addPayment (IPayment p){
        paymentMethod = p;
    }
}
Public class BankPayment implements IPayment {
    public bool pay(){
        /* ... code ... */
    }
}
Public class TestPayment implements IPayment {
    public bool pay (){
        /* ... code ... */
    }
}

```

#	Klausimas	Balai
8	Papildykite diagramą, pritaikant Adapter šabloną taip, kad atsiskaitymą grynais (COD) (angl. Cash On Delivery) galima būtų naudoti kaip mokėjimo metodą. Susiekite atitinkamais ryšiais.	1,5

Pseudokodas

```
class PaymentAdapter implements IPayment {  
  
    private COD cod;  
  
    public bool pay() {  
        cod.cash();  
    }  
}
```

T120B029 EGZAMINO PAVYZDYS