BBM102 - Spring 2020 - Midterm 1

Dosyaları yükleyip bu formu gönderdiğinizde Google hesabınızla ilişkilendirilen ad ve fotoğraf kaydedilir.

Question 1

Please answer the following true/false questions. Write a brief explanation (a few sentences) for each question. (8 points)

a) Encapsulation provides the code changeability.

Yanıtınız

b) A class inherits from multiple classes directly by using the “extends” keyword.

Yanıtınız

c) Composition does not promote the software reusability, but inheritance does.

Yanıtınız

d) We could define two different types of instance variables with the same name in a class.

Yanıtınız

BBM102 - Spring 2020 - Midterm 1

Dosyaları yükleyip bu formu gönderdiğinizde Google hesabınızla ilişkilendirilen ad ve fotoğraf kaydedilir.

oyunoynama113@gmail.com değil misiniz? Hesap değiştir

Question 2

Please give the class definitions and the relations between classes (inheritance/composition) for the following scenario: Ten teams compete in the champions league every year. Each team has a game strategy to win. There are two game strategies based on attack or defense. You do not need to write any method/constructor of the classes. (10 points)

Yanıtınız

Sayfa 3 / 8

Geri Sonraki

Google Formlar üzerinden asla şifre göndermeyin.

BBM102 - Spring 2020 - Midterm 1

Dosyaları yükleyip bu formu gönderdiğinizde Google hesabınızla ilişkilendirilen ad ve fotoğraf kaydedilir.

Question 3

Please answer the following true/false questions. Write a brief explanation for each question. (10 points)

Question 3, code

```
class A {
        private int i;
        public  int x;
        public void f() { /* ... */ }
        public void g() { /* ... */ }
}

class B extends A {
        public int j;
        public void g() { /* ... */ }
}

class C extends B {
        public int k;
        public void g(int x) {/*.....*/}
}

...
C c = new C();
B b = new B();
A a = new A();
...
```

a. We could access the variable j directly from the method g of Class C.

Yanıtınız

b. We could access the variable i directly from the method g of Class B.

Yanıtınız

c. c.g() is a valid statement.

Yanıtınız

d. We could call the method g of Class A from the method g of Class B by using the following statement: g()

Yanıtınız

e. c.x = 3 is a valid statement

Yanıtınız

Geri Sonraki

BBM102 - Spring 2020 - Midterm 1

Dosyaları yükleyip bu formu gönderdiğinizde Google hesabınızla ilişkilendirilen ad ve fotoğraf kaydedilir.

oyunoynama113@gmail.com değil misiniz? Hesap değiştir

Question 4

Define the early binding and late binding. Explain the advantages and disadvantages of each binding method. (6 points)

Yanıtınız

Geri Sonraki

BBM102 - Spring 2020 - Midterm 1

Dosyaları yükleyip bu formu gönderdiğinizde Google hesabınızla ilişkilendirilen ad ve fotoğraf kaydedilir.

Question 5

Please answer this question regarding the code block given below: (20 points)

Question 5, code

```
Line | Code
 1  public class Student {
 2     protected int no;
 3     public String name;
 4     public int beginningYear;
 5
 6     public Student() {
 7        no = -1;
 8     }
 9     public void setNo(int pNo) {
10        System.out.println ("No number is assigned!");
11     }
12     public int getNo() {
13        System.out.println ("Number cannot be read!");
14        return -1;
15     }
16  }
17  public class UnderGradStudent extends Student{
18
19     private String supervisor;
20
21     public UnderGradStudent () {
22        super();
23     }
24     public void setNo(int pNo) {
25        no = pNo;
26        System.out.println (no + " is assigned.");
27     }
28     public int getNo() {
29        System.out.println (no + " is read.");
30        return no;
31     }
32  }
33
34     public static void main(String args[]) {
35        int myNo;
36        Student student = new Student();
37        UnderGradStudent underGradStudent = new UnderGradStudent ();
38        student.setNo(256);
39        underGradStudent.setNo(421);
40        myNo = student.getNo();
41        student = underGradStudent;
42        myNo = student.getNo();
43     }
```

Please give the row number(s) on the program that corresponds to these concepts if they exist in the code. You must write which concepts are matching with the code lines that you selected. Please write an explanation for that line. (E.g. If there is polymorphism why is there on that line?)

1. Inheritance:

Yanıtınız

2. Encapsulation:

Yanıtınız

3. Overriding:

Yanıtınız

4. Overloading:

Yanıtınız

5. Polymorphism:

Yanıtınız

Geri Sonraki

BBM102 - Spring 2020 - Midterm 1

Dosyaları yükleyip bu formu gönderdiğinizde Google hesabınızla ilişkilendirilen ad ve fotoğraf kaydedilir.

Question 6

In this question, you are given a scenario given below and expected to provide an easily viewable and understandable class diagram ("diagram.png" or "diagram.jpg" file) and its corresponding single file code ("code.txt", i.e. you will not provide multiple code file. Try to collect them in just a single file). You must provide each of these files. However, in case you cannot provide, you will gain a score for your answered parts. Please read the scenario carefully. For the diagram, you can use any diagram tool supporting UML class diagrams. (46 points)

Scenario:

In this question, we design a factory that produces cellphones for various vendors such as Samsung, Apple, or Nokia. For the sake of simplicity, we consider cellphone production with two important components: (1) battery and (2) screen.

In this scenario, we have two different types of batteries such as LiON battery and NiMH battery. As is known, NiMH batteries are capable of storing less energy than liON batteries and they do not support fast recharging. To denote the maximum energy for the battery itself, we utilize the variable of "capacityAmp" that holds integer values(e.g. 3000 for NiMH, 5000 for LiON). Moreover, the property of "isFastRecharging" is another important property that can store binary information. In order to learn information about the battery, each kind of battery must implement a method named "diagnose". This method is responsible for returning a string-based information about the capacity and the fast recharging property of the created battery.

Similarly, any cellphone in our example factory can either have an LCD Screen or AMOLED Screen. These screens differ from each other in terms of an integer "PPI" property - pixel per inch - (e.g. 400 or 540) and supported resolutionInfo – a String based information (e.g. "1920x1080" or "3200x1440"). Note that, the AMOLED screen supports higher resolution and higher PPI values. Similar to battery object, screen objects must implement a method named "diagnose" to inform users about the PPI and resolution of the assembled screen.

Similarly, any cellphone in our example factory can either have an LCD Screen or AMOLED Screen. These screens differ from each other in terms of an integer "PPI" property - pixel per inch - (e.g. 400 or 540) and supported resolutionInfo – a String based information (e.g. "1920x1080" or "3200x1440"). Note that, the AMOLED screen supports higher resolution and higher PPI values. Similar to battery object, screen objects must implement a method named "diagnose" to inform users about the PPI and resolution of the assembled screen.

Apart from these, the factory named "CellPhoneFactory" must implement just one single method named "produceCellPhone" with appropriate parameters. This class is responsible for the production (i.e assembling the parts) and returning the CellPhone object checking the following rules:

*Nokia phones cannot have AMOLED Screen
*Apple phone cannot have LCD screen
*Samsung phone cannot have NiMHBattery

According to the above constraints, this method either successfully creates a phone and returns it, otherwise it returns a null reference along with a failure prompt on the screen.

The main function of the main method is given below:

```
public static void main(String[] args) {

    Battery nimhBattery =  new NiMHBattery(3000,false);
    Battery lionBattery = new LiONBattery(5000,true);
    Screen lcdScreen = new LCDScreen(400,"1920x1080");
    Screen amoledScreen = new AMOLEDScreen(540,"3200x1440");
    CellPhoneFactory cpf = new CellPhoneFactory();
    CellPhone cphone1 = cpf.produceCellPhone("Nokia","X", nimhBattery, lcdScreen);
    System.out.println( cphone1.getInfo());
    CellPhone cphone2 = cpf.produceCellPhone("Samsung","S7",nimhBattery,lcdScreen)
    CellPhone cphone3 = cpf.produceCellPhone("Apple","5S",nimhBattery,lcdScreen);
    CellPhone cphone4 =
cpf.produceCellPhone("Samsung","S20Ultra",lionBattery,amoledScreen);

}
```

The following output is produced after executing this main method:

```
Factory has produced one Nokia-X
Info: Nokia X[The battery does not support fast recharge having max 3000 mAH - The
screen has max resolution of 1920x1080px at 400PPI]
Samsung phone cannot have NiMHBattery
Apple phone cannot have LCDScreen
Factory has produced one Samsung-S20Ultra
```

Question 6 Files (code)

Dosya ekle

Question 6 Files (diagram)

Dosya ekle

Geri Sonraki