

Norges Informasjonsteknologiske Høgskole

Skriftlig prøve

PG2100 – Programmering 2

Tillatte hjelpemidler: ingen

Dato: 3.6.14

Vedlegg som kan være aktuelle: 12 (side 5 – 18)

Tid: 180 minutter

I alle oppgavene teller hvert delspørsmål likt dersom ikke annet er oppgitt.

NB! Hvis du synes noe er uklart eller at opplysninger mangler, må du gjøre egne begrunnede antagelser/forutsetninger, og løse oppgaven ut fra disse.

Oppgave 1 (35 %)

Klassen `GroceryItem` representerer en vare i en handlekurv. Klassen har tre attributter, to konstruktører og standard tilgangsmetoder. I tillegg har den en `toString`-metode og en `equals`-metode – se skisse under.

```
public class GroceryItem {  
    private String name;  
    private int quantity;  
    private double pricePerUnit;  
  
    //konstruktører  
  
    //tilgangsmetoder  
  
    //equals  
  
    //toString  
  
}
```

a) (5 %) Skriv konstruktøren som har parametere for de tre attributtene. Konstruktøren skal bruke tilgangsmetodene.

b) (5 %) Skriv `toString`-metoden.

c) (10 %) Skriv `equals`-metoden.

Vedlegg 1 viser klassen `GroceryList`. Den representerer en liste med varer (`GroceryItem`'s). Listen er implementert som en array av type `GroceryItem`.

d) (15 %) Skriv en ny versjon av klassen `GroceryList` der varelisten er implementert som en `ArrayList` med passende type, i stedet for en array.

Oppgave 2 (35 %)

Skriv en applikasjon som fungerer som en valutakalkulator – se figur under:

NOK	
GBP	EUR
USD	SEK
Resultat	
Avslutt	

Brukeren skriver inn et beløp i norske kroner, og kalkulatoren regner om til ønsket valuta. I figuren under har brukeren skrevet inn beløpet 1000 NOK og klikket EUR (valgt omregning til euro), og kalkulatoren viser resultatet.

NOK	1000
GBP	EUR
USD	SEK
Resultat	NOK 1000.0 = 133,37 EUR
Avslutt	

Ved klikk på en av de andre valutaknappene vil omregningen skje i forhold denne valutaen. Ved klikk på knappen "Avslutt" avsluttes applikasjonen.

Skriv koden for denne applikasjonen.

Du kan bruke følgende valutakurser:

- 1 GBP = 8.873 NOK
- 1 USD = 5.6 NOK
- 1 EUR = 7.498 NOK
- 100 SEK = 84.880 NOK

Oppgave 3 (30 %)

Følgende klasser er definert:

<pre>public class Biggie extends JayZ { public void a() { System.out.print("Biggie a "); super.a(); } public String toString() { return "Biggie"; } } public class JayZ extends Tupac { public void a() { System.out.print("JayZ a "); b(); } }</pre>	<pre>public class FiftyCent extends Biggie { public void b() { System.out.print("FiftyCent b "); } } public class Tupac { public void a() { System.out.print("Tupac a "); } public void b() { System.out.print("Tupac b "); } public String toString() { return "Tupac"; } }</pre>
---	---

a) (10 %) Gitt klassene over – hva blir output når følgende program blir utført?

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

        Tupac[] elements =
            {
                new Biggie(),
                new Tupac(),
                new JayZ(),
                new FiftyCent()
            };

        for (int i = 0; i < elements.length; i++) {
            elements[i].a();
            System.out.println();
            elements[i].b();
            System.out.println();
            System.out.println(elements[i]);
            System.out.println();
        }
    }
}
```

Følgende klasse representerer en høgstolestudent:

```
public class Student {  
    private String name;  
    private int age;  
  
    public Student() {}  
  
    public Student (String name, int age) {}  
  
    public void setName(String name) {}  
  
    public void setAge(int age) {  
        this.age = age;  
    }  
  
    public String getName() {}  
  
    public int getAge() {}  
}
```

Under er det vist en delvis definisjon av klassen UndergraduateStudent, som er en subklasse til klassen Student.

```
public class UndergraduateStudent extends Student {  
    private int year;  
  
    public UndergraduateStudent() {}  
  
    public void setYear(int year) {}  
  
    //...  
}
```

- b) (5 %) Kan kode i denne klassen aksessere (bruke direkte) attributtet `age` arvet fra `Student`? Forklar.
- c) (5 %) Kan kode i denne klassen kalle metoden `setAge` arvet fra `Student`? Forklar.
- d) (10 %) Skriv en konstruktør for klassen `UndergraduateStudent` som har parametre for `name`, `age` og `year`, og som gir parameternes verdier til de tilsvarende attributtene på en passende måte.

--- Slutt på oppgavesettet ---