

Tentamen i

Grundläggande programmering

STS, åk 1

lördag 2002-05-25

Skrivtid: 09.00 – 14.00

Hjälpmedel: Inga

Lärare: Anders Berglund. Elena Fersman besöker tentan vid två tillfällen: cirka kl. 10.30 samt cirka kl. 11.30. Anders Berglund är inte anträffbar under tentan.

Observera:

- Programkod ska skrivas enligt god programmeringssed för Java och objektorienterad programmering.
- Uppgifterna är inte ordnade efter svårighetsgrad.
- Läs igenom hela tentan först.
- Kom ihåg att fylla i försättsbladet ordentligt.
- Om du är osäker på någon fråga eller uppfattar någon uppgift som oklar, gör ett antagande som du motiverar och förklarar väl.
- På de uppgifter där du ska skriva programkod, kan du också skriva en förklaring, om du inte är säker på att din lösning är riktig. En sådan ger visserligen inga poäng, men kan hjälpa till vid rättningen att förstå tankarna bakom en lösning
- Skriv programkod rimligt indenterad och med bra variabelnamn. På så sätt underlättas rättningen.
- 50% av tentans totala poängantal garanterar betyget godkänt
- Skriv endast en uppgift per blad och skriv bara på ena sidan av papperet, dvs. ett nytt blad för varje uppgiftsnummer, medan deluppgifter (a, b, c etc.) får skrivas på samma blad.
- Fyll i försättsbladet ordentligt!
- Skriv namn på alla papper.
- Referera inte mellan olika svar.

Lycka till!

FÖRSÄTTSBLAD

Denna sida ska ifylld placeras först bland de papper du lämnar in

Uppgift	Besvarad ange i förekommande fall vilka deluppgifter du har löst
1	
2	
3	
4	
5	
6	
7	
bonus	<i>ifylles av institutionen</i>

Namnteckning: _____

Namn: _____

Personnummer: _____

1. Ord, bara ord (4p)

Förklara kortfattat med precist följande termer:

- a) syntax (i samband med programmeringsspråk)
- b) semantik (i samband med programmeringsspråk)
- c) parameter (i samband med programmeringsspråk)
- d) lättviktig komponent (i samband med grafik i Java)
- e) tungviktig komponent (i samband med grafik i Java)
- f) komposition (i samband med programmeringsspråk)
- g) klassmetod (i samband med programmeringsspråk)
- h) Applet (i samband med Java)

2. Det snurrar (6p)

Vilken utskrift ger nedanstående program? Om du anser att programmet avbryts med ett felmeddelande, går i en evig loop eller något sådant, ska du även kort förklara varför

```
import extra.*;

public class GorVadKlass{

    public static int GorVad(int x) {
        if (x == 0) return 1;
        if (x == 1) return 1;
        return GorVad(x-1) + GorVad(x-2);
    }

    public static void main(String[] args) {
        for (int i = 5; i <= 8; i++) {
            Std.out.println(GorVad(i));
        }
    }
}
```

3. OOP (6p)

Ange kortfattat tre väsentliga argument till varför objekt-orienterade programmeringsspråk är lämpliga för mycket programutveckling.

4. Personlig integritet (4p)

Under föreläsningen i dataetik behandlades begreppet personlig integritet. Ge exempel på en kränkning av den personliga integriteten och förklara kortfattat varför den är en kränkning. Ge din förklaring i högst fyra meningar, gärna kortare

5. Tabeller (20p)

Läs hela uppgiften, innan du påbörjar någon deluppgift!

Studera nedanstående programskelett. På ett antal ställen saknas kod. Din uppgift är att lösa nedanstående deluppgifter på ett sådant sätt, att lösningarna passar in i programskelettet och med det provkörningsexempel som finns sist i uppgifter.

Programmet ska fråga först efter ett ensamt tal och sedan ytterligare ett antal tal. De inmatade talen ska skrivas ut. Därefter ska alla förekomster av det första talet i de övriga inmatade talen tas bort. Till sist ska alla tal skrivas ut igen.

- a) I denna deluppgift ska du skriva de rader som saknas i metoden `in()`, så att metoden, när den anropas, läser in ett antal tal till `minTabell`. Inmatningen ska avslutas med 0. Du ska i denna uppgift använda en `while`-slinga.
- b) I denna deluppgift ska du skriva de rader som saknas i metoden `ut()`, så att metoden, när den anropas, skriver ut de element som är skilda från 0 (noll). Du ska i denna deluppgift använda en `for`-loop.
- c) I denna deluppgift ska du skriva de rader som saknas i metoden `remove(int x)`. Metoden ska ta bort alla förekomster av talet `x` ut arrayen. Du avgör själv vilka programspråkskonstruktioner du vill använda.
- d) I denna deluppgift ska du skriva de rader som saknas i huvudprogrammet. Körexemplet visar vad huvudprogrammet ska göra. Skriv huvudprogrammet mycket enkelt.

Du får utnyttja att samtliga tal i en array, när den skapas, automatiskt av Java sätts till 0. Du får också anta att användaren aldrig kommer att försöka mata in mer än ett litet antal tal - aldrig 100 stycken tal, eller ett antal, ungefär lika stort som 100 (såsom 98 eller 99). (Detta antagande kan vara till hjälp beroende på hur du väljer att lösa uppgiften).

forts på nästa sida

forts från föregående sida

```
import extra.*;

public class TabellKlass {
    private int[] minTabell = new int[100];

    public void in() {
        Std.out.println("Ge ett antal tal. Avsluta med 0.");
        int tp = Std.in.readInt();

        Här saknas kod

    }

    public void skriv() {

        Här saknas kod

    }

    public void remove(int x) {

        Här saknas kod

    }

    public static void main(String[] args) {
        TabellKlass mittObjekt = new TabellKlass();
        Std.out.println("Ge det tal som ska tas bort");
        int bort = Std.in.readInt();

        Här saknas kod

    }
}
```

Provkörning:

```
Ge det tal som ska tas bort
5
Ge ett antal tal. Avsluta med 0.
6
Ge nästa tal
5
Ge nästa tal
3
Ge nästa tal
4
Ge nästa tal
5
Ge nästa tal
6
Ge nästa tal
0
Talen före
6 5 3 4 5 6
Talen efter
6 3 4 6
```

6. *Bilar* (20p)

Tyvärr har du gett upp dina studier vid STS programmet för att börja som bilhandlare. Dock, du vill ha nytta av vad du har lärt dig och vill datorisera lagret av begagnade bilar. I detta syfte skriver du en klass `Bil`, som beskriver en begagnad bil i lagret. Varje bil har ett märkesnamn och ett modellnamn (båda får lagras som strängar). För varje bil finns det ett inpris (det du som bilhandlare betalar för bilen) och ett utpris (det pris du säljer bilen för). Utpriset beräknas från inpriset, med en procentsats. Normalt är denna 120% (dvs. en bil som du har köpt för 10 000.- säljer du för 12 000.-). Ibland är det dock sommarrea och procentsatsen sjunker, eller vårrusning, då du höjer priserna.

Alla förändringar i priserna sker genom ändringar av denna procentsats. Procentsatsen är densamma för alla bilar. Det innebär att när du ändrar procentsatsen så förändras priset för alla bilar.

Skriv nu denna klass.

För att få full poäng på uppgiften, måste du skriva klassen med

- de variabler som finns i klassen
- en konstruktor
- en metod som skriver ut all information om en bil, inklusive de båda priserna
- en metod som låter dig ange en ny procentsatsen
- ett enkelt huvudprogram, som använder dessa metoder. Huvudprogrammet behöver inte ha texter som hjälper användaren, vara genomtänkt etc., utan är enbart tänkt att kunna fungera som ett provkörningsexempel för dessa metoder.

Du behöver inte skriva några andra metoder än de som anges. Du får göra all in- och utmatning enkel. Tänk på att programmet ska illustrera en god programmeringssed för funktionella språk.

7. *Klasser och sån't* (6p)

Studera nedanstående ord:

Magsäck, hund, kattdjur, häst, däggdjur, djur, talgoxe, svans, husse/matte, leopard, tamkatt

Relatera orden till varandra med hjälp av begreppen *arv*, *komposition*, *känner till*.

Om du vill, kan du skriva en förklaring till de relationer du har beskrivit. Skriv i så fall förklaringen på ett separat papper. En sådan förklaring ger inga poäng, men kan underlätta rättningen.

Glad sommar!!