

## Lösningsförslag, lösningsansatser

### Tentamen 2006-01-20, Grundläggande programmering i Java

#### Uppgifter

Läs skrivningsanvisningarna och följ dessa! Överträdelse har menlig inverkan på bedömning

- 1 . Beskriv exekveringen av en javaapplikation. Hur skiljer den sig från en vanlig applikation?

4 p

**Lösning:**

Lösningen ska minst innehålla

- javaprogramkod kompileras till javabytekod
- javabytekoden kan köras av en javatolk
- javatolken är specifik för en speciell plattform/operativsystem
- en vanlig applikation kan bara köra på en plattform, en java applikation kan i princip köras på alla javatolkar/plattformar

2. Hur skiljer sig typerna **double** och **Double**?

3 p

**Lösning:** **double** är en enkel typ som representerar ett värde. **Double** är en klass som innehåller bland annat hjälpfunktioner till typen double; i synnerhet String-> double omvandling

Not: Flera var inne på Scannerklassens metod nextDouble(), next**Double**() har ingenting med **Double** att göra, det är två helt olika namn. På samma sätt som System.out.**print** inte har någonting med typen **int** att göra. Det versala D i nextDouble kommer ju av namngivningskonventionen.

3. Beskriv utförligt skillnaden mellan en klass och ett objekt.

3 p

**Lösning:** En klass är en beskrivning av ett objekt. Då en klass instansieras skapas ett objekt, med hjälp av en konstruerare. Objekt har en (klass)typ, klasser är en typ.

4. Vilka namngivningskonventioner för identifierare finns i Java? Beskriv konventionen för medlemmar och klasser.

2 p

Vad innebär "camel case"?

**Lösning:** Medlemmar - inleds med gemen, därefter camelCase. Klasser - inleds med Versal därefter CamelCase. CamelCase innebär att varje ord avstavas med versal. VarjeOrdAvstavasMedVersal.

En konvention är en gemensamma överenskommelser som inte är implementerade i syntaxregler. Om det är ord man inte förstår på en tenta har man möjligheten att ställa frågor om frågan (naturligtvis aldrig om lösningen). Det finns som regel ordböcker och liknande till hjälp också.

5. Vad blir utskriften av följande kompletta program? Motivera

4 p

```
import java.text.*;  
public class Main{
```

```
public static void main(String [] a){
    double sum = 0;
    for(int i=0;i< 100000;i++)
        sum += Math.random();
    double result = sum/100000;
    DecimalFormat df=new DecimalFormat("0.00");
    System.out.println( df.format( result ) );
}
}
```

Det är inte en slamkrypare! Det finns ett svar till uppgiften!

**Lösning:**

Satsen `sum += Math.random()` itereras 100000 gånger. `Math.random()` ger ett värde mellan 0(inklusive) och 1(exklusive). Alltså måste `sum` efter iterationen ha ett värde som är större än 0 och mindre än 100000. Efter divisionen är värdet mellan 0 och 1. (denna del ger max två poäng)

Ett slumpvärde kan liknas vid en slantsingling, det är lika vanligt att krona kommer upp som att klave kommer upp, så efter 100000 slantsinglingar har ungefär lika många krona som klave kommit upp, säg 50000 vardera. Om man tilldelar krona värdet 0, och klave värdet 1, max och min på slumpvärdet, blir summan alltså ungefär 50000 (mindre än +/- 300 de flesta gångerna, för de som läst statistik på högskolan). Summar summerar, medelvärdet av flera slumpantal ligger troligtvis i mitten av intervallet, det behövs ingen matematik för att inse det, man måste däremot tänka sig en körning av programmet och höfta värden på `Math.random()`. Utskriften blir 0.50. Denna betydligt svårare del av uppgiften ger max två poäng.

6. Skriv ett fullständigt javaprogram som producerar följande användardialog och utskrift:

8 p **Hur många rader?**

```
6
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
```

Användaren har matat in heltalet 6, och programmet skriver därför sex rader.

**Lösning:**

Kärnan i lösningen är två for-satser, där den yttre for-satsen ger den inre for-satsen maxvärde. Som

```
for(int rad=1;rad<=antalRader;rad++){
    for(int kolumn=1;kolumn<=rad;kolumn++){
        System.out.print(kolumn+" ");
    }
}
```

Eller att man lägger till motsvarande siffra för varje rad. Som

```
String värden = "";
for( int i = 1;i<= antalRader;i++){
    värden = värden + i + " ";
    System.out.println( värden );
}
```

7. Betrakta programavsnittet

4 p

```
Scanner keyb = new Scanner(System.in);
int i = keyb.nextInt();
if ( villkor ){
    System.out.println("Villkoret är sant!");
}
```

Skriv villkor så att villkoret är sant om

a) heltalet **i** ligger i intervallet, från och med 1 till och med 3.

**Lösning: ( i >=1 && i<=3)**

b) heltalet **i** ligger utanför intervallet från och med 1 till och med 3.

**Lösning: ( i<1 || i>3 )**

c) heltalet **i** är udda.

**Lösning: ( i%2 == 1 ), eller (i%2 != 0)**

8. Beskriv steg för steg, algoritmen för sorteringsmetoden selection sort.

5 p **Lösning:** se anteckningar

9. Skriv ett komplett program som skriver ut antalet textrader i en fil på skärmen.  
8 p Textfilens namn är C:\texts\largefile.txt

Du ska i din lösning använt klasserna FileReader och BufferedReader. Om du glömt bort vilka metoder som klasserna innehåller, så gissa. En rimlig gissning ger färre poängdrag. Glöm inte try / catch. Undantaget som ska fångas är av typen IOException.

**Lösning:**

Lösningen innehåller följande steg (med klassdeklaration, main och deklarerationer)

```
try {  
    FileReader fr = new FileReader("c:\\texts\\largefile.txt");  
    BufferedReader reader = new BufferedReader( fr );  
    while( reader.readLine() != null )  
        nLines++;  
} catch( IOException ioe){}  
System.out.println( nLines );
```

10. Följande är ett utdrag ur dokumentationen av metod i klassen JOptionPane

4 p

<pre>static String</pre>	<pre>showInputDialog(Component parentComponent, Object message)</pre> <p>Shows a question-message dialog requesting input from the user parented to parentComponent.</p>
--------------------------	--

Förklara utförligt vad det står. Ge utifrån din förklaring ett exempel på hur metoden kan anropas.

**Lösning:**

Frågan gäller vad som står i utdraget, ingenting annat.

statisk metod - man använder Klassnamn.showInput... för att anropa metoden.

returnerar ett Stringvärde - man har ett objekt av typen string i vänsterledet vid en tilldelning av metodens returvärde; String s = JOptionPane.ShowInput

Component parentComponent - Detta argument ger inga poängdrag. Anger vilket fönster som är förälder till meddelanderutan. Används för att positionera fönstret.

Object message - Anger vilket meddelande/fråga som ska visas i rutan.

Ex: String answer = JOptionPane.showInputDialog( null, "Vad är svaret?");

11. Vilken funktion har klassen JScrollPane?

2 p Måste man instantiera klassen för att den ska vara användbar? Varför?

**Lösning:** Att förse olika komponenter med Rullningslister, ex en JEditorPane. Ja, klassen måste instansieras. Av flera anledningar - JScrollPane har inte static-medlemmar för att påverka rullningslisterna - Varje rullningslist är oberoende av andra rullningslister - I NetBeans ger man JScrollPane-komponenten ett unikt namn. alltså är det ett objekt.

12. Beskriv funktionen layouthanteraren BorderLayout

2 p **Lösning:** En BorderLayout har 5 ytor, NORTH, SOUTH, WEST, EAST och CENTER. Med hjälp av borderlayout kan man placera in (max) 5 komponenter, i en yta avsedd för en komponent.