

PROGRAMMERING-Java

Omtentamina

Nicolina Månsson
2007 - 08 - 13

Tentamensinstruktioner

Poängsättning

Hela tentamen omfattar 41 poäng. Poäng för varje uppgift står angivet inom parentes före varje uppgift.

- För att få betyget 3,4 eller 5 krävs det minst 20,28 eller 37 poäng
- För att få betyget G eller VG krävs det minst 20 eller 32 poäng

Redovisning

I varje uppgift står det vad som skall göras för att få angivet antal poäng.
Redovisa koden i den ordning den efterfrågats i texten med så mycket implementerat som möjligt.
Tänk på att vara noggrann och strukturerad. Det är Du som skall visa vad Du kan!

Hjälpmedel

Finns hos tentavakten.

Lycka till!

Uppgift 1 - Enkla program med variabler, metoder, val och repetition satser (3p+3p+3p+3p+3p)

a) Skriv ett enkelt program som slumpar 3 heltalsvärden i intervallet 1-6. Med en while-loop skall programmet slumpa repeterade gånger tills alla 3 heltalen som slumpas råkar ha samma värde. Då skall programmet avslutas och skriva ut hur många gånger loopen har gått.

Ledning ! Använd en räknare-variabel som uppdateras för varje iteration i loopen. Slumpa tal med hjälp av ett Random-objekt. Konstrueraren från klassen Random och metoden nextInt() som ska användas för att slumpa tal är definierade som nedan:

	<code>Random ()</code> Creates a new random number generator.
<code>int</code>	<code>nextInt(int n)</code> Returns a pseudorandom, uniformly distributed <code>int</code> value between 0 (inclusive) and the specified value (exclusive), drawn from this random number generator's sequence.

b) Skriv ett program som från en användare läser in 4 heltalsvärde (x1,y1) och (x2,y2), som representerar koordinaterna för två punkter. Därefter skall programmet beräkna och skriva ut avståndet mellan de två punkterna. Formeln du skall använda är:

$$\text{avstånd} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} .$$

Använd metoderna från klassen Math:

<code>static double</code>	<code>sqrt(double a)</code> Returns the correctly rounded positive square root of a double value.
<code>static double</code>	<code>pow(double a, double b)</code> Returns the value of the first argument raised to the power of the second argument.

c) För att konvertera en vinkel från grader till radian multipliceras graderna med PI och delas med 180.0 , alltså:

`double radians = (degrees * Math.PI)/180.0 ;`

Skriv ett program som använder ett loop och skriver ut en tabell med vinklar från **0** till **360** både i grader och radian.

d) Använd en while-loop i ett program som läser in positiva heltal från användaren och skriver ut den största av dem. Loopen avbryts efter 10 inläsningar eller om ett negativ tal matas in.

Tips! Använd en variabel (kalla den max) . För varje loop jämför max med den nya inmatade värdet .

e) Skriv ett program som läser in ett heltal från användaren. Om inmatat tal är mellan 0 och 9 ska "Ensiffrigt" skrivas ut utom om talet är 7 då istället "Mitt lyckotal" skrivas ut. Är talet mellan 10 och 99 skall "Tvåsiffrigt" skrivas ut utom om talet är 77 då "Dubbel lycka" skrivs ut. Om talet är negativ eller 3-siffrigt skall "Utanför intervallet" skrivas ut.

Uppgift 2 – Klasser, metoder och Arrayer, (4p+ 8p+2p+4p)

I följande klasser beskrivs ett "telefonregister" i vilket man kan lagra namn och telefonnummer för ett antal personer.

Så, för att skapa ett telefonregister av personer måste du först implementera klassen Person. En Person i detta sammanhang består enbart av namn och telefonnummer, som du ser nedan. Implementera färdig klassen enligt beskrivningen som finns framför varje metod.

```
class Person {  
  
    private String namn;  
    private String telNr;  
  
    /** skapa en person med namnet name och telefonnumret telNr */  
    public Person(String name, String telNr);  
  
    /** returnerar personens namn*/  
    public String getNamn();  
  
    /** returnerar personens telefonnummer*/  
    public String getTelNr();  
}
```

b) Ett telefonregister är helt enkelt en array av Person-objekt. Att implementera metoder för insert och remove i och från en array av Person-objekt är mycket likt det du har gjort på laborationen där du har lagt till och tagit bort heltal från en array (klassen IntSet). Implementera klassen TelefonKatalog enligt nedanstående beskrivning.

```
class TelefonKatalog {  
  
    private Person[] lista;  
    private int counter;  
  
    /** konstruerare, skapa arrayen lista med plats för 100  
    personer*/  
    public TelefonKatalog();  
  
    /** lägg in en person (p) i listan*/  
    void insert(Person p);
```

```

/** tag bort personen med namnet name ur listan. Tänkt på att du
först måste söka och hitta platsen där personen med respektive
namn finns i listan */
void delete(String name);

/** returnera telefonnumret för personen i listan som har namnet
name (null om ingen sådan person finns) */
public String findTelefon(String name);

}

```

c) Finns det några nackdelar med att implementera telefonregistret som en array. Vilka? Vad finns det för annat alternativ?

d) Skriv **ett program** som använder klassen Person och TelefonKatalog och gör följande:

1. Skapar ett TelefonKatalog-objekt.
2. Skapar två Person-objekt.
3. Söker efter ett telefonnummer genom att anropa metoden findTelefon(). Skriver ut telefonnumret eller skriver ut "Inget telefonnummer för namnet X" om ingen person med respektive namn hittas.

Uppgift 3 - (8p)

a) Komplettera nedanstående program så att den fungerar som ett enkel miniräknare. Som du ser kan användaren mata in till programmet en sträng innehållande ett enkelt aritmetiskt uttryck (som t.ex 3+5 eller 56*23). Programmet skall utifrån den inmatade strängen beräkna uttrycket.

```

public class Calculator
{
public static void main (String [] arg)
{
String st1, String st2; // variabler som skall lagra värde i strängform.
int ta1, int ta2; // variabler som skall lagra värde i heltalsform
char operator=''; // variabeln ska lagra den hittade operatör
int plats= -1; // variabeln skall lagra positionen i strängen där operatören hittades
Scanner scan =new Scanner (System.in);
System.out.println("Mata in uttryck : " );
String str = scan.next();

// hitta platsen i strängen där någon av tecken + eller * eller - eller / finns och
// spara respektive tecken, använd metoden charAt() och indexOf()
// dela uttrycket i två mindre strängar, använd metoden substring()
// konvertera sträng talen till heltalsvärde, använd metoden parseInt()
// gör beräkning utifrån den hittade operatören och skriv ut resultatet
}
}

```

Ledning: Använd någon/några av följande metoder från klassen `String` och `Integer`.
Obs! Konvertering från sträng till heltal skall göras efter det att har delat den inmatade stängin sträng uttrycket i de två tal och en operator.

int	<code>length()</code> Returns the length of this string.
int	<code>indexOf(String str)</code> Returns the index within this string of the first occurrence of the specified substring.
String	<code>substring(int beginIndex, int endIndex)</code> Returns a new string that is a substring of this string.
char	<code>charAt(int index)</code> Returns the char value at the specified index.

Konvertering av sträng till heltal skall göras med metoden `parseInt()` som finns i klassen **Integer** och definieras som nedan.

static int	<code>parseInt(String s, int radix)</code> Parses the string argument as a signed integer in the radix specified by the second argument.
------------	---------------------------------------------------------------------------------------------------------------------------------------------