

Kontrollskrivning 1, Programmeringsteknik för D/C

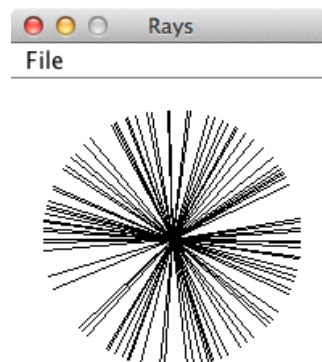
2013–10–16, 10.30–11.45

Anvisningar:

- Fyll i omslaget fullständigt, även lösta uppgifter och antalet inlämnade blad.
- Skriv bara på en sida av varje papper.
- Lämna bara in dina lösningar och omslaget, inte skrivningen och inte några kladdpapper.
- Tillåtet hjälpmedel: Java-snabbpreferens.

Jag meddelar på kurshemsidan när rättningen är klar (någon gång i tentamensveckan). Dina poäng på skrivningen anslås på institutionens anslagstavla (E-huset andra våningen, höger uppgång). Där meddelar jag också när du kan hämta ut din skrivning.

-
1. Skriv ett program som ritar en bild med följande utseende i ett `SimpleWindow`-fönster med storleken 200×200 pixlar:



- Man tänker sig en cirkel med mittpunkten i fönstrets mittpunkt och radien 80 pixlar.
- Man har ritat 100 linjer.
- Varje linje går från cirkelns mittpunkt till en slumpmässig punkt på cirkelns periferi. (Dra en slumpmässig vinkel och räkna ut punktens x - och y -koordinater.)
- Man skapar fönstret med `new SimpleWindow(200, 200, "Rays")`.
- `SimpleWindow`-operationen `moveTo(x,y)` flyttar pennan till punkten (x,y) utan att rita, operationen `lineTo(x,y)` ritar en linje.
- I `SimpleWindow` finns också operationerna `getWidth()` och `getHeight()` för att ta reda på fönstrets bredd och höjd.

Vänd!

2. En mäklarfirma har ett antal kolonistugor till försäljning. Stugorna identifieras med unika id-nummer. Mäklarfirman har ett register över de bud som lagts på stugorna. Implementera klassen Bid, som beskriver ett bud:

```

/** Skapar ett budobjekt. Budet gäller stugan med nummer houseId, är på
    amount kronor och är lagt av kunden bidder */
Bid(int houseId, int amount, Customer bidder);

/** Tar reda på stugans idnummer */
int getId();

/** Tar reda på budets storlek (i kronor) */
int getAmount();

/** Tar reda på kunden som lagt detta bud */
Customer getBidder();

/** Returnerar skillnaden mellan detta bud och budet b (i kronor). Resultatet
    ska bli ett positivt tal om detta bud är högre än budet b */
int getDifference(Bid b);

```

En kund beskrivs av den färdigskrivna klassen Customer, som har följande specifikation:

```

/** Skapar en kund med namnet name */
Customer(String name);

/** Tar reda på kundens namn */
String getName();

```

3. Senare i kursen ska vi se hur man gör för att hålla reda på många objekt av samma slag. Nu nöjer vi oss med att se på ett litet exempel som skapar och använder några budobjekt. Studera följande program:

```

public class SmallBidExample {
    public static void main(String[] args) {
        Bid b1 = new Bid(27, 230000, new Customer("Lisa Alm"));
        Bid b2 = new Bid(27, 200000, new Customer("Bo Ek"));
        int diff = b1.getDifference(b2);
        Bid highest;
        if (diff >= 0) {
            highest = b1;
        } else {
            highest = b2;
        }
        System.out.println("Högsta budet är " + highest.getAmount()
            + " och budgivare är " + highest.getBidder().getName());
    }
}

```

- Vilka variabler deklarerats i main-metoden? Ange både namn och typ.
- Hur många objekt skapas? (Räkna inte med String-objekt och liknande.)
- Vad skriver programmet ut?
- Rita en figur som visar vilka variabler och objekt som finns efter det att if-satsen exekverats. Ange också värdet av alla variabler och attribut. Ange värdena av referensvariabler med pilar. Inuti klassen Customer finns ett attribut name.