Lösningsförslag till delmålskontroll, EDA501 Programmering 2015-02-27

Rättningsanvisningar

Skriv **era namn följt av ordet rättare** på omslaget på den/de lösningar ni rättar. Använd en **penna av avvikande färg** så att det tydligt framgår vad som är era rättningskommentarer.

Markera för det som är fel och rätta felet om möjligt. Skriv också en kort kommentar om vad som är fel. Efter varje lösning finns en checklista som kan vara till hjälp vid rättningen.

Kommentera även sådant som inte är fel, men ändå olämpligt. Ex svårläst programkod där ordentliga indragningar inte har gjorts, konstiga eller intetsägande variabelnamn. Sätt dock ingen bock i dessa fall.

När ni rättat klart så **markera tydligt poängen på omslaget** (skriv poängen och ringa in den t ex) enligt:

2 poäng: Om ni tycker att alla tre deluppgifterna är behandlade minst ok*.
1 poäng: Om två av deluppgifterna är behandlade minst ok* eller en av deluppgifterna är behandlad riktigt bra**.
0 poäng: I övriga fall.

*) Behandlad **ok** betyder att det får finnas en del fel, men naturligtvis inte hur många som helst.
) Behandlad **riktigt bra betyder att det högst får finnas något litet fel.

1.
```
public class DiceGame {
   public static void main(String[] args) {
      Die d = new Die();
      int max = 0;    // Mindre än 1
      int min = 7;    // Större än 6
      for (int roll = 1; roll <= 5; roll++) {
         d.roll();
         int eyes = d.getResult();
         if (eyes > max) {
            max = eyes;
         }
         if (eyes < min) {
            min = eyes;
         }
      }
      int diff = max – min;
      System.out.println("Största differens: " + diff);
   }
}
```

- Är ”ramen” runt programkoden rätt:
```
public class DiceGame {
    public static void main(String[] args) {
       ...
    }
 }
```
- Klassen Die behöver inte importeras. Vi förutsätter att alla färdiga klasser finns tillgängliga när vi löser uppgifter på papper.
- Är alla variabler deklarerade innan de används? Har de rätt typer? Variabelnamn bör börja med liten bokstav.
- Skapas ett Die-objekt i början?
- Är variablerna max och min initierade korrekt innan repetitionen?
- Stämmer for-satsen? Är det rätt start- och slutvärden? Finns { } omkring satserna som skall utföras i repetitionen?
- OBS! Det måste vara två olika if-satser för kontrollen av max och min. Det kan inte vara en if-else-sats då första tärningsslaget kommer bli både max och min.
- Hanteras tärningsslagen på rätt sätt – d.roll() och d.getResult()?
- Stämmer uträkningen av differensen?
- Är den avslutande utskriften korrekt? Finns det citationstecken omkring teckensträngar? Kontrollera att det inte finns citationstecken omkring variabelnamn när variabelns värde skall skrivas ut.
- Är programmet korrekt för övrigt? Löses rätt problem?

2.
```
public class Rook {
   private int x;     // position i x-led
   private int y;     // position i y-led

   /** Skapar ett schacktorn som befinner sig i positionen (x,y)
       på ett schackbräde. */
   public Rook(int x, int y) {
      this.x = x;
      this.y = y;
   }

   /** Tar reda på tornets position i X-led (horisontellt). */
   public int getX() {
      return x;
   }

   /** Tar reda på tornets position i Y-led (vertikalt). */
   public int getY() {
      return y;
   }

   /** Flyttar tornet n steg (n positivt eller negativt) i X-led.
       Du får förutsätta att tornet inte hamnar utanför brädet. */
   public void moveAlongX(int n) {
      x = x + n;
   }

   /** Flyttar tornet n steg (n positivt eller negativt) i Y-led.
       Du får förutsätta att tornet inte hamnar utanför brädet. */
   public void moveAlongY(int n) {
      y = y + n;
   }

   /** Avgör om det är tillåtet att flytta tornet till positionen (x,y).
       Du behöver inte ta hänsyn till andra pjäser på schackbrädet och
       du får förutsätta att (x,y) är inom brädet. */
   public boolean potentialPosition(int x, int y) {
      if (x==this.x && y==this.y) {             // Tornet kan inte flytta
         return false;                          // dit det redan står
      } else if (x==this.x || y==this.y) {      // Om kolumn ELLER rad
         return true;                           // samma
      } else {
         return false;
      }
   }
}
```

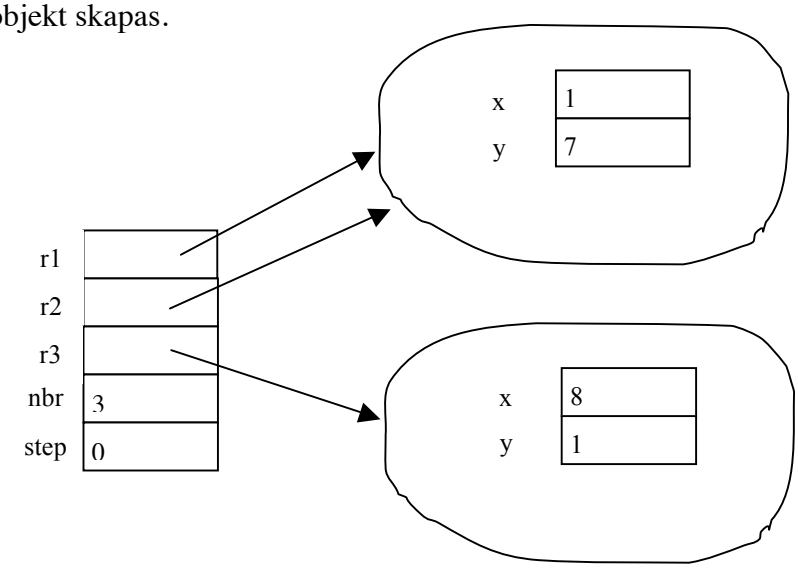
- Saknas något attribut? Finns det något onödigt attribut med (något som istället borde vara deklarerat lokalt inuti en metod)? Har attributen rätt typer? Det måste finnas attribut för både x- och y-led.
- Är det med private eller public deklarerat vad som skall vara åtkomligt utanför klassen eller ej?
- Görs rätt saker i konstruktorn? Får attributen rätt startvärden? Observera att attribut automatiskt får startvärden när de skapas (gäller inte andra variabler). (Attribut av typen int får startvärdet 0, attribut av typen referens får startvärdet null etc.). Men i det här fallet tilldelas nog alla attribut andra värden i konstruktorn.
- Görs rätt saker och används rätt attribut i respektive metod.
- Skiljer man på attribut och parameter/lokal variabel när/om de har samma namn, dvs används this.x för att komma åt attributet.
- Finns det en return-sats i varje funktion (get-metod) och är den korrekt? Returneras rätt värde? Stämmer returvärdets typ med typen som anges i metodrubriken.
- Är x-led horisontellt? Är y-led vertikalt?
- Om man i metoden potentialPosition har glömt testa fallet med att tornet inte kan flyttas till den positionen den redan står på så bedöm bara det som ett litet fel.

3. a) 1,7
1,7
8,1

b) r1, r2, r3, nbr och step.

c) Två objekt skapas.

d)



- Finns alla variabler med?
- Variabeln i skall inte finnas med (den finns bara tillfälligt i for-satsen).
- Refererar referensvariablerna till rätt objekt?
- Finns rätt antal objekt med (2 st)? Objekt skapas bara med new.
- Finns alla attribut med i objekten? Tänk på att antal attribut och namnen på attributen skall stämma överens med den skrivandes lösning till uppgift 2.
- Har attributen rätt värden?