



Antal blad /  
Number of sheets

12 ✓

# TENTAMEN / EXAMINATION

- Anvisningar:** Skriv din anonymitetskod på varje blad.  
Endast en uppgift får lösas på varje blad.  
Var vänlig skriv tydligt!
- Instructions:** Write your anonymous code on each sheet.  
Answer only one question on each sheet.  
Please write clearly!

Vänligen texta anonymitetskoden i textboxen enligt exempel nedan!  
Please write the Anonymous Code clearly in the textbox like example below!

**Bokstäver/Letters:**

A-B-C-D-E-F-G-H-I-J-K-L-M-N-O  
P-Q-R-S-T-U-V-W-X-Y-Z-Å-Ä-Ö

**Siffror/Numbers:**

Ø-1-2-3-4-5-6-7-8-9

Exempel:

A B C 1 7 Ø - Ø 1 7

DVGA02 Programutvecklingsmetodik  
Kurskod + Kurs / Course Code + Course:

Delkurs / Part course:

Anonymitetskod / Anonymous code =  
Kurskod + kodnr / course code + code number  
DVGAØ2 - Ø29

Tentamensdatum /  
Examination date:  
2017/01/16 ✓

## Behandlade uppgifter / Solved problems

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
X	X	X	X	X	X	X	X							
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

## Ifylles av lärare / To be completed by the examiner

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
20	3	2.5	3	5	2	3	3							
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

Poäng / Marks gained: 22

Betyg / Grade: 3

Max poäng / Total marks gained: 40

För Gk poäng / Marks gained to be passed: 20

Examin. lärare / Kursansvarig signatur / Signature of the examiner

Namnförtydligande / Clarification of the signature



1

a, Skillnaderna kan jämföras med en ritning och det faktiska huset. Alltså klassen beskriver hur en sak ska se ut och vad den kan göra. Samt att objektet är en instans av den ritningen. Alltså ett objekt av det klassen beskriver.

1,5

b, Man borde använda inbäddning för att göra så att allting inte hänger på varann och om man måste ta bort eller ändra någon liten del så påverkar inte allt. Även saker som att de vet varje kliss sin privata egenskaper och det enda sättet att få reda på dem är att fråga dem, med tex en getter.

1



1

c, En iterator är t.ex,  
en for loop. En fördel  
är att

En nackdel är att då  
måste bestämma innan  
exakt hur många varu  
den ska köras.

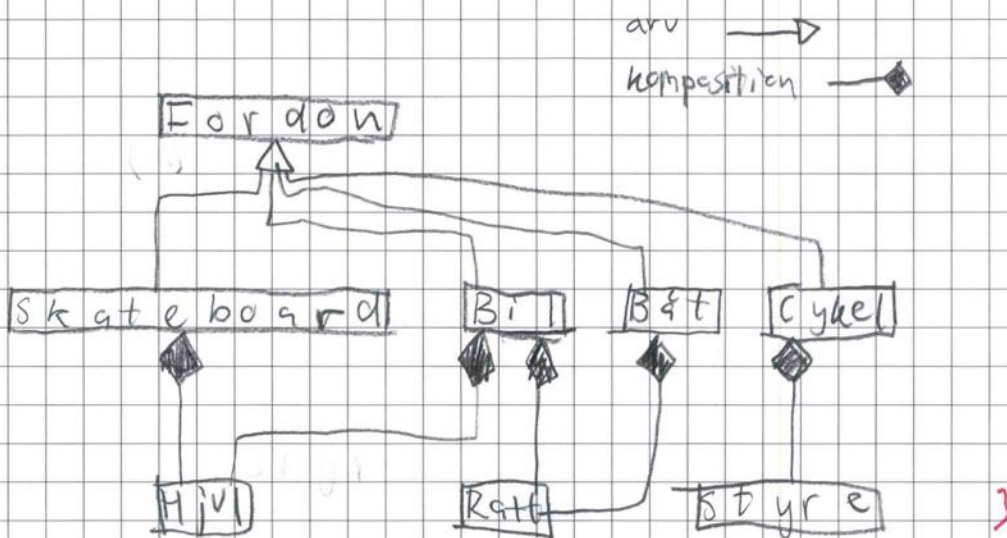
0





2, skillnaden mellan arv  
a, och komposition är stor

När någonting ärver från  
något annat så får den alla  
attributer och egenskaper och alla  
föräldraklassens metoder. När en sak ärver  
från någon annan är den den  
och lite till. Vid komposition  
består någonting av något annat  
t.ex. en tröja och tröden.



b,  
Komposition för klassen AAA består  
av ett objekt av BBB, samt att  
den vet vad BBB kan göra.



3, public interface Åtbar

{

private int kalorier

public bliÅten() {

}

}

public class Ost implements Åtbar {

{

public bliÅten() {

this.destroy();

}

public class Ägg implements Åtbar

{

public bliÅten() {

this.crack();

this.destroy();

}

SA här kan gränssnitts polymorfism  
se ut. Med hjälp av detta kan  
vi göra operationen på objekten av  
klasserna beroende på om de  
är ätbara eller inte.



Ange anonymitetskod / Write your anonymity code  
(Vid icke anonym tentamen ange kurskod + namn + personnummer)  
(For non-anonymous exams write the course code + name + civic registration number)

DUGA02-029

Löpande sidnr  
Consecutive no:

5

Uppgift nr /  
Question no:

3

Poäng / Points  
awarded:

2.5

Lärarens  
anteckning  
Examiner's remarks:

Man kan jobba med dom  
mer allmänt och inte behöva  
veta vilken av Ågg eller Öst det  
är utan bara ifall dom är giltiga  
kan vi hantera dom på det sättet.

Ett exempel på anteckning kunde varit b.a.

2.5





4

a, ActionListener är ett gränssnitt som man implementerar om man t.ex. vill kunna trycka på en knapp och nått händer. När man implementerar ActionListener måste man även implementera actionPerformed och skriva vad den ska göra. actionPerformed hanterar saker och beroende på vad som hänt gör den olika saker. addActionListener kör man på objekt som man vill lyssna på. Alltså t.ex. en knapp.

2

b

*ActionEvent*

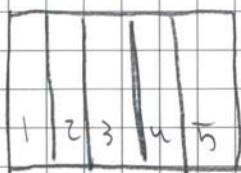
```

actionPerformed(Element E){
    E.getSource() == button1
    if (button1) System.exit(0);
    else if (button2) ball.setDx(ball.getDx()*2);
    else ball.move();
  
```

0.5



C, En flow layout ser ut så här



Där man antingen  
räknar ifrån höger  
eller vänster för att

positioner a ut säker i rutan.

Här går jag från vänster och det  
är bara att välja vilken plats  
och lägga in komponenten på,

Bestäm om ordningen redan  
ligger rätt i.

0.5





5

Pre: Sant

Post: valklista med minimum värder

Valklista();

Pre: size &lt; MAXIMUM

Post: size++, element inlagd

LaggaTill(Person P, int index);

pre: size &gt; MINIMUM och personen finns i listan

post: size--, element borttaget

TaBort(Person P);

Pre: size &gt; MINIMUM och index ≤ size

Post: size--, element borttaget

TaBort(int index);

Pre: Sant

Post: returnerar hur många personer

som finns i listan

size();

Pre: Sant

Post: returnerar true om size = MAXIMUM

returnerar false om size &lt; MAXIMUM

ArFull();

Bra ADT

fast rule

eller Java listor

3



Ange anonymitetskod / Write your anonymity code  
(Vid icke anonym tentamen ange kurskod + namn + personnummer)  
(For non-anonymous exams write the course code + name + civic registration number)

DVGA02-029

Löpande sidnr  
Consecutive no:

9

Uppgift nr /  
Question no:

5

Poäng / Points  
awarded:

Lärens  
anteckning  
Examiner's remarks:

Pre: sant

Post: returnerar true om  $size = MINIMUM$

returnerar false om  $size > MINIMUM$

AtMinimum();

Häftområde

Skriv ej i detta område  
Leave this area blank



6 Referenser i Java är samma sak  
som pekare i C, skillnaden  
är bara att vi har inte lika  
fri tillgång till dem. I Java  
vill dem sköta referenserna "under  
hooden" men i C att vi fria  
till att bruka och missbruka  
pekare hur mycket vi vill.

Minuskaraktär, syntax, GC, etc





Ange anonymitetskod / Write your anonymity code  
(Vid icke anonym tentamen ange kurskod + namn + personnummer)  
(For non-anonymous exams write the course code + name + civic registration number)

DVGA02-029

Löpande sidnr  
Consecutive no:

11

Uppgift nr /  
Question no:

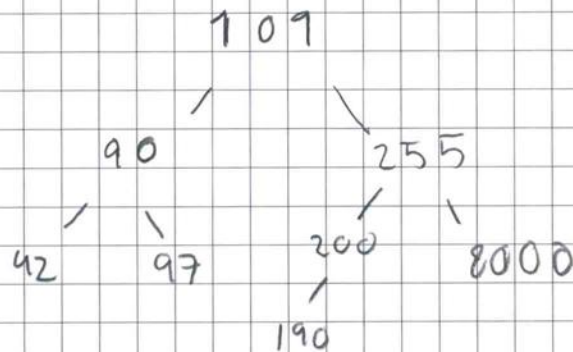
7

Poäng / Points  
awarded:

Lärarens  
anteckning  
Examiner's remarks:

7

a,



b,

```

addElement (int data) {
  if (data > root) {
    addhöger(data)
  }
  else {
    addvänster(data)
  }
}

```



8

9,

```
public class Sprite {
```

```
    private int x;
```

```
    private int y;
```

```
    private int width;
```

```
    private int height;
```

```
    private int dx;
```

```
    private int dy;
```

```
    private Color color;
```

b

```
    move();
```

```
{
```

```
        setX(getX() + getDx());
```

```
        setY(getY() + getDy());
```

```
}
```

Behöver 4 st  
hoket

0.5

c,

```
    collision(Sprite s)
```

```
{
```

```
    if (this.intersects(s)) {
```

```
        this.setDy(getDy() - 1);
```

```
        this.setDx(getDx() - 1);
```

```
}
```

1.5