



Antal blad /
Number of sheets

07 ✓

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
---	---	---	---	---	---	---	---	---	---

ISGB18 Introduktion till objektorienterad programmering

Kurskod + Kurs / Course Code + Course:

Delkurs / Part course:

Anonymitetskod / Anonymous code =
Kurskod + kodnr / course code + code number

I	S	G	B	1	8	-	Ø	Ø	5
---	---	---	---	---	---	---	---	---	---

 ✓

Tentamensdatum /
Examination date:

10/3-17

Behandlade uppgifter / Solved problems

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

Poäng / Marks gained:

39

Betyg / Grade:

V4

Max poäng / Total marks gained:

40

För Gk poäng / Marks gained to be passed:

20

Examin. lärare / Kursansvarig signatur / Signature of the examiner

Johan Öfverberg

Namnförtydligande / Clarification of the signature



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)

15GB18-005

Löpande sidnr
Consecutive no:

1

Public class Tenta {

Uppgift nr /
Question no:

1

Poäng / Points
awarded:

5

Lärarens
anteckning
Examiner's remarks:

Public static void main(String[] args) {

int i;

int h=100;

for(i=0; i<21; i++){

System.out.println(i + "\t" + h);

h = h - 5;

} // stänger for-loop

} // stänger main

} // stänger class.



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)

ISGB18-005

Löpande sidnr
Consecutive no:

2

Uppgift nr /
Question no: 2

Poäng / Points
awarded:

3

Lärarens
anteckning
Examiner's remarks:

"Nu i Pgm1" - Super.skriv(s) anropar skriv()-metoden i superclass Pgm1.

"Nu i Pgm2" - Programmet går tillbaka till class Pgm2 & kör sista raden i metoden skriv()

"Nu i Pgm1" - Det skapas en ny Pgm2 med referens till Pgm1, vilket är möjligt pga arv. Super.skriv() anropas.

"Nu i Pgm2" - Tillbaka i class Pgm2 kör sista raden i metoden skriv().

"Nu i Pgm1" - referens objektet ändras till att vara en instans av Pgm1. Därför kör endast skriv()-metoden i klassen Pgm1



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)

ISGB18-005

Löpande sidnr
Consecutive no:

3

Uppgift nr /
Question no: 3

Poäng / Points
awarded:

Lärarens
anteckning
Examiner's remarks:

a) Klass är en ritning på hur något ska se ut och hur det ska fungera. Ett objekt är en instans av en klass, man har använt ritningen och "byggt" något.

Om variabler eller metoder är static kan man komma åt dem utan att skapa en instans, genom Klass.metod(), tex Integer.parseInt().

3

b) Polymorfism är att klasser ärver beteende från en annan. Sen är det upp till subclassen att metoden fungerar på rätt sätt för just den subclassen. tex.

```
public class Djur { public void ätMat() {} }
public class Ko extends Djur {
    public void ätMat() {
        gåTillGräs();
        tuggaGräs; } }
```

3

```
Public class Lejon extends Djur {
    Public void ätMat() {
        jagaByte();
        ätByte(); } }
```

c) Aggregat är en relation där den ena klassen "äger" den andra klassen. Tex Lektor & fakultet. Lektorn "ägs" av fakulteten, men försvinner fakulteten finns lektorn kvar... Det är en en-vägs relation.

Fortsättning →



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)

1SGD18-005

Löpande sidnr
Consecutive no:

4

Uppgift nr /
Question no:

3

Poäng / Points
awarded:

10

Lärarens
anteckning
Examiner's remarks:

c) Lektor har 1 fakultet, Fakultet kan ha många Lektor.

Composition → Lektor döms med Fakultet.

Association → En fakultet lånar en lektor som
inhoppare från en annan fakultet.

2

Fakultet f = new Fakultet(new Lektor());

d) Association är när två klasser
känner till varandra och kan interagera
med varandra, tex att en klass tar
emot ett referensobjekt via in-parametern.

Aggregat är när relationen bara är
en en-vägs relation. En student måste
ha en adress, MEN, en adress måste inte
ha en student.

2



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)

ISGB18-005

Löpande sidnr
Consecutive no:

5

Uppgift nr /
Question no:

4

Poäng / Points
awarded:

8

Lärarens
anteckning
Examiner's remarks:

```
a) public class Cirkel {
    private double r;
    public Cirkel(double d) {
        r = d;
    }
    public double getRadien() {
        return r;
    }
}
```

```
public double getOmkrets() {
    return (r * 3.14);
}
```

/* Jag har gjort antaganden att man alltid skickar in ett värde som inte ger error. */

```
b) public static void main(String[] args) {
    Cirkel c = new Cirkel(5.5);
    System.out.println("Radien: " + c.getRadien());
    System.out.println("Omkrets: " + c.getOmkrets());
}
```

Skriv ej i detta område
Leave this area blank



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)

1SGB18-005

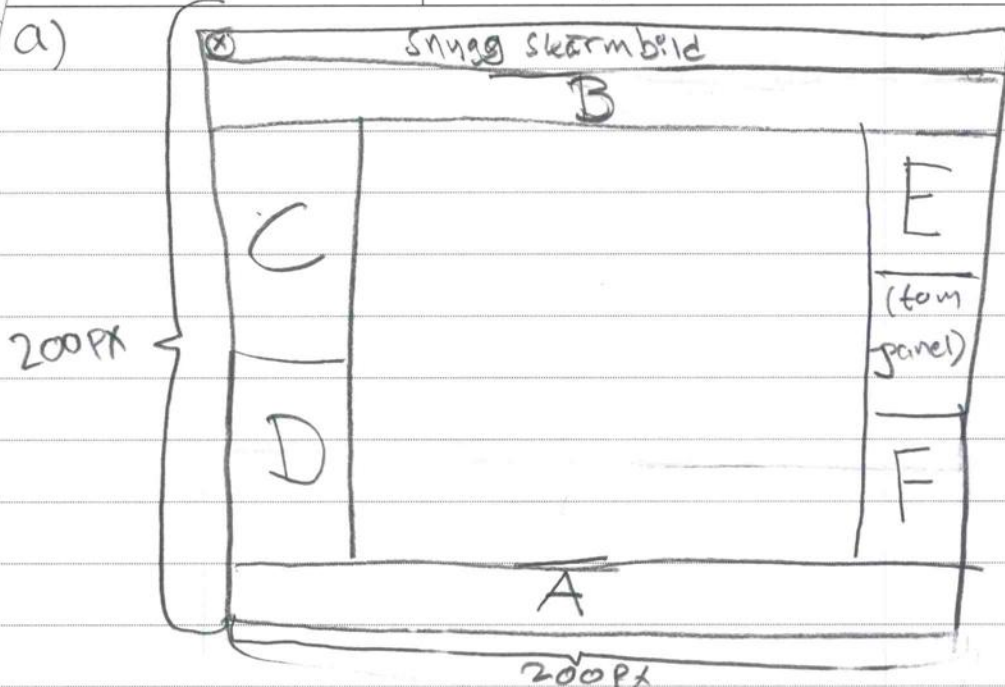
Löpande sidnr
Consecutive no:

6

Uppgift nr /
Question no: 5

Poäng / Points
awarded:

Lärarens
anteckning
Examiner's remarks:



B) I klass-kroppen "Private Lyssnare l;
I konstruktor: "l = new Lyssnare(this);"
i klassen lägg till "private JTextArea t;"
i konstruktorn innan f.setVisible() skriv
"t = new JTextArea();" och efter det:
"f.add(t, BorderLayout.CENTER);"
I konstruktor efter "b2 = new JButton("B");" &
efter vi skapat ett objekt av Lyssnare-klass.
"b.addActionListener(l);"
Efter "b5 = new JButton("E");" skriv:
"b5.addWindowListener(this);"
Efter att JFrame har skapats:
"f.addWindowListener(this);"
I klass-kroppen utanför main & konstruktor
skriv denna metod:
public void skrivText() {
t.setText("Hej"); } }

Fortsättning nästa blad.



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)

ISGBB-005

Löpande sidnr
Consecutive no:

7

Uppgift nr /
Question no:

5

Poäng / Points
awarded:

13

Lärarens
anteckning
Examiner's remarks:

B) import java.awt.event.*; // IS
Public class Lyssnare extends WindowAd-
apter implements ActionListener {
Private Grafik g;
Public Lyssnare (Grafik g) {
this.g = g; }

Public void actionPerformed(ActionEvent e) {
g.skrivText();
// Onödigt att kolla vilken JButton som
// trycktes för vi har bara 1 med
// actionListener.
}

Public void windowClosing(WindowEvent w)
{ System.exit(0);
}
} // stänger class.

Skriv ej i detta område
Leave this area blank