

## **i Info**

**UNIVERSITETET I OSLO**

**Det matematisk-naturvitenskapelige fakultet**

**Skriftlig midtveiseeksamen i IN1010**

**2024 vår**

**Varighet: 18. mars kl 9:00 til 11:00**

**Tillatte hjelpemidler:**

Ingen

### **Annen informasjon**

Faglærerne kommer *ikke* til å besøke eksamenslokalet under eksamen. Om du mener at det er noe *alvorlig* galt med en oppgave, kan du ta kontakt med en eksamensvakt.

Alle oppgavene er uavhengig av hverandre og kan løses i vilkårlig rekkefølge.

Det finnes en kalkulator tilgjengelig; se link nederst på skjermen.

# 1 Enkel-programmering

Sett kodebitene på riktig plass så det blir et lovlig og kjørbart program.

Du får poeng for riktig linje på rett plass, men 0 poeng for feil.

 Hjelp

```
class Pakke {
  int vekt;
  Pakke (int v) { vekt = v; }
}
```

```
class LagPakke {
```

```
public static void main (String[] args) ✓
```

```
  Pakke min, din; ✓
```

```
  min = new Pakke(5); ✓
```

```
  din = min; ✓
```

```
  int vekten = din.vekt; ✓
```

```
  System.out.println(vekten); ✓
```

```
  }
```

```
}
```

```
min = new Pakke(5);
```

```
din = min;
```

```
public static void main (String[] args) {
```

```
System.out.println(vekten);
```

```
Pakke min, din;
```

```
int vekten = din.vekt;
```

Maks poeng: 12

## 2 Arv+komposisjon

Når er det naturlig å bruke arv og når er det naturlig med komposisjon i disse relasjonene?

Du får poeng for hvert riktig svar, minuspoeng for hvert galt svar og null poeng hvis du ikke gjør et valg. Du vil aldri få mindre enn 0 poeng totalt.

**NB!** Inspira er slik at om du først har gjort et valg, kan du forandre valget, men du kan ikke lenger velge å ikke gjøre et valg.

	Komposisjon	Arv
Sete – sykkel	<input type="radio"/> ✓	<input type="radio"/>
Kjøretøy – sykkel	<input type="radio"/>	<input type="radio"/> ✓
Hjul – sykkelhjul	<input type="radio"/>	<input type="radio"/> ✓
Bil – taxi	<input type="radio"/>	<input type="radio"/> ✓
Terrengsykkel – sykkel	<input type="radio"/>	<input type="radio"/> ✓
Bil – hjul	<input type="radio"/> ✓	<input type="radio"/>
Person – professor	<input type="radio"/>	<input type="radio"/> ✓
Bok – illustrasjon	<input type="radio"/> ✓	<input type="radio"/>
Bokhandel – bok	<input type="radio"/> ✓	<input type="radio"/>
Butikk – bokhandel	<input type="radio"/>	<input type="radio"/> ✓

Maks poeng: 5

### 3 Enkel-lesing

Lag et program **MinstAv5** som leser fem ikke-negative heltall og skriver ut det minste.

Du får poeng for riktig kodebit på rett plass men 0 poeng for feil. Ikke alle kodebitene skal brukes.

 Hjelp

```
import java.util.Scanner;
class MinstAv5 {
    public static void main (String[] args) {
        Scanner tastatur = new Scanner(System.in);
        int min = 0 ;
        for ( int i = 1; i <= 5; i++ ) {
            int v = tastatur.nextInt()
            if ( i == 1 || v < min ) {
                min = v ;
            }
        }
        System.out.println( min );
    }
}
```

int i = 1; i <= 5; i++

min == i

i

min

min = 0

int i = 1; i < 5; i++

int v = tastatur.nextInt();

i == 1 || v < min

i = 5

v

min = v

MinstAv5

Maks poeng: 10.01

## 4 Person-objekter

```

class StatiskPerson {
    public static void main (String[] args) {
        Person eva = new Person("Eva", 3);
        Person ole = new Person("Ole", 11);
        eva.økTall(4);
        ole.økTall(-1);
    }
}

class Person {
    int tall1;
    static int tall2;
    String navn;

    Person(String id, int n) { navn = id; tall1 = n; tall2 = 0; }

    void økTall (int n) {
        tall1 = tall1 + n;
        tall2 = tall2 + (2 * n);
        System.out.print(navn + ": " + tall1);
        System.out.println(" og " + tall2);
    }
}

```

Hva skrives ut av dette programmet?

**NB!** Legg merke til at *tall2* er definert som **static**!

Du får poeng for hvert riktig svar og 0 poeng for feil.

Eva:  (7) og  (8)

Ole:  (10) og  (6)

---

Maks poeng: 12

## 5 Pekersjonglering

```
class Person {
    String navn;
    Person (String n) { navn = n; }
}

class LagEiere {
    public static void main (String[] args) {
        Person eva = new Person("Eva");
        Person ole = new Person("Ole");
        Person per = new Person("Per");
        Person meg, du, nabo, vennen, kollega;

        meg = eva;
        du = ole;
        nabo = per;
        kollega = du;
        vennen = eva;
        du = vennen;
        System.out.println("Nr 1: " + du.navn);
        System.out.println("Nr 2: " + meg.navn);
        System.out.println("Nr 3: " + kollega.navn);
    }
}
```

Velg hva som skrives ut.

Du får poeng for riktig svar og minuspoeng for galt svar, men du vil aldri få mindre enn 0 poeng totalt. Velger du intet alternativ, får du 0 poeng på den delen.

**NB!** Hvis du har valgt et alternativ, kan du endre det, men du kan aldri gå tilbake til å velge ingenting.

## Hvilke navn skrives ut?

	Eva	Ole	Per
Nr 1	<input type="radio"/> ✓	<input type="radio"/>	<input type="radio"/>
Nr 2	<input type="radio"/> ✓	<input type="radio"/>	<input type="radio"/>
Nr 3	<input type="radio"/>	<input type="radio"/> ✓	<input type="radio"/>

Maks poeng: 9

## 6 Katt+mus+ost



```

class KattOgMusOgOst {
    public static void main (String[] args) {
        Katt tutsi = new Katt(3000);
        Katt mons = new Katt(4000);
        Katt graapus = new Katt(5000);

        Mus pip = new Mus(200);
        Mus lille = new Mus(300);
        Mus smal = new Mus(400);

        Ostebit minst = new Ostebit(10);
        Ostebit liten = new Ostebit(20);
        Ostebit stor = new Ostebit(30);

        pip.spis(minst);
        lille.spis(liten);

        mons.spis(pip);
        tutsi.spis(lille);
        graapus.spis(smal);

        System.out.println("Tutsi veier: " + tutsi.vekt);
        System.out.println("Mons veier: " + mons.vekt);
        System.out.println("Gråpus veier: " + graapus.vekt);
    }
}

class Katt {
    int vekt;
    Katt (int v) { vekt = v; }
    void spis (Mus m) { vekt = vekt + m.vekt; }
}

class Mus {
    int vekt;
    Mus (int v) { vekt = v; }
    void spis (Ostebit ost) { vekt = vekt + ost.vekt; }
}

class Ostebit {
    int vekt;
    Ostebit (int v) { vekt = v; }
}

```

Hva skrives ut av dette programmet?

Du får poeng for hvert riktig svar og 0 poeng for feil.

Tutsi veier:  (3320)

Mons veier:  (4210)

Gråpus veier:  (5400)

---

Maks poeng: 12

## 7 Frekvensteller

```
class TestFrekvens {
    public static void main (String[] args) {
        Frekvens frek = new Frekvens(4);
        for (int i = 0; i<=4; i++) {
            frek.observert(i);
            frek.observert(i+1);
            frek.observert(i+2);
        }
        frek.observert(2);
        frek.observert(4);

        for (int ind = 0; ind < frek.hentAntall(); ind++) {
            System.out.println("Tall nr " + ind + ": " + frek.hentVerdi(ind));
        }
    }
}

class Frekvens {
    private int[] tall;

    Frekvens (int ant) { tall = new int[ant]; }

    void observert (int indeks) {
        if (indeks >= 0 && indeks < tall.length)
            tall[indeks] = tall[indeks] + 1;
    }

    int hentAntall () { return tall.length; }
    int hentVerdi (int indeks) { return tall[indeks]; }
}
```

Hvilke tallverdier skrives ut av dette programmet?

Du får poeng for riktig svar og 0 poeng for galt.

Tall nr 0:  (1)

Tall nr 1:  (2)

Tall nr 2:  (4)

Tall nr 3:  (3)

---

Maks poeng: 12

## 8 Rekursiv-utskrift

Vi har en generell listeklasse med en metode *skrivBaklengs* som skal sørge for at alle elementene i listen blir skrevet ut *i omvendt rekkefølge*, dvs at noden som *start* peker på, skal skrives ut sist. Metoden skal gjøre dette ved å kalle på den **rekursive** metoden *skrivRekursivt*.

Sett inn kodebitene på riktig plass. Ikke alle kodebitene skal brukes.

Du får poeng for riktig kodebit på rett plass, og 0 poeng for feil.

 Hjelp

```
class Liste <T> {
    class Node {
        Node neste;
        T data;

        void skrivRekursivt () {
            if ( neste != null ) {
                neste.skrivRekursivt();
            }
            System.out.println(data.toString());
        } // slutt på metoden skrivRekursivt
    } // slutt på indre klasse Node
    Node start = null;

    void skrivBaklengs () {
        if ( start != null ) {
            start.skrivRekursivt();
        }
    }
}
```

start.skrivRekursivt()

data != null

skrivBaklengs()

neste != null

neste != this

start != null

neste.skrivRekursivt()

neste != start

System.out.println(data.toString())

System.out.println(neste)

start.skrivBaklengs()

skrivRekursivt(neste)

this != null

---

Maks poeng: 10

## 9 Polymorfe-metoder

```

class Polymorfi {
    public static void main (String[] args) {
        Vare[] alleVarer = new Vare[4];
        alleVarer[0] = new Vare(3);
        alleVarer[1] = new Matvare(7);
        alleVarer[2] = new Nødmat(12);
        alleVarer[3] = new IkkeMat(20);

        for (int i = 0; i < 4; i++) {
            System.out.println("Bruttopris " + i + ": " +
                alleVarer[i].bruttoPris());
            System.out.println("Vanlig pris " + i + ": " +
                alleVarer[i].pris);
        }
    }
}

class Vare {
    int pris;
    Vare (int v) { pris = v; }
    public int bruttoPris () { return pris * 2; }
}

class Matvare extends Vare {
    Matvare (int v) { super(v); }
    @Override public int bruttoPris() { return pris; }
}

class Nødmat extends Matvare {
    Nødmat (int v) { super(v); }
    @Override public int bruttoPris() { return pris / 2; }
}

class IkkeMat extends Vare {
    IkkeMat (int v) { super(v); }
}

```

Dette programmet skriver ut 8 verdier. Hvilke verdier skrives ut?

Du får poeng for hvert riktig svar og 0 poeng for feil.

Bruttopris 0:  (6)

Vanlig pris 0:  (3)

Bruttopris 1:  (7)

Vanlig pris 1:  (7)

Bruttopris 2:  (6)

Vanlig pris 2:  (12)

Bruttopris 3:  (40)

Vanlig pris 3:  (20)

---

Maks poeng: 8

## 10 Referansetilordning

```

interface Egglegger {
    int antallEgg ();
}

class Hare {
    final String navn;
    Hare (String n) { navn = n; }
}

class Påskehare extends Hare implements Egglegger {
    int antall;
    Påskehare (String n, int ant) { super(n); antall = ant; }

    @Override public int antallEgg () { return antall; }
}

class Høne implements Egglegger {
    @Override public int antallEgg () { return 1; }
}

class BrukHare {
    public static void main (String[] args) {
        Hare skogshare = new Hare("Bjørk");
        Hare sesonghare = new Påskehare("Påsan", 999);
        Høne burhøne = new Høne();

        // Hvilke av disse 10 linjene er lovlige som neste linje?
        Påskehare h1 = sesonghare;
        Egglegger h2 = sesonghare;
        Egglegger h3 = burhøne;
        Egglegger h4 = (Egglegger)skogshare;
        Påskehare h5 = (Påskehare)sesonghare;
        Egglegger h6 = (Hare)sesonghare;
        Hare h7 = sesonghare;
        Egglegger h8 = (Egglegger)sesonghare;
        Egglegger h9 = skogshare;
        Påskehare h10 = (Påskehare)skogshare;
    }
}

```

Dette lille programmet tester tilordning og typekonvertering av referanser til objekter. Av de ti indikerte linjene, hvilke

- kompileres og kjøres uten problemer?
- vil gi kompileringsfeil eller feil under kjøring?

Riktig svar vil gi poeng og galt svar minuspoeng, men du vil aldri få mindre enn 0 poeng totalt.



Om du er usikker, kan du velge å ikke svare, og da får du 0 poeng. **NB!** Inspira er slik at om du først har gjort et valg, *må* du velge et av de to alternativene.

	OK	Feil
Påskehare h1 = sesonghare;	<input type="radio"/>	<input type="radio"/> ✓
Egglegger h2 = sesonghare;	<input type="radio"/>	<input type="radio"/> ✓
Egglegger h3 = burhøne;	<input type="radio"/> ✓	<input type="radio"/>
Egglegger h4 = (Egglegger)skogshare;	<input type="radio"/>	<input type="radio"/> ✓
Påskehare h5 = (Påskehare)sesonghare;	<input type="radio"/> ✓	<input type="radio"/>
Egglegger h6 = (Hare)sesonghare;	<input type="radio"/>	<input type="radio"/> ✓
Hare h7 = sesonghare;	<input type="radio"/> ✓	<input type="radio"/>
Egglegger h8 = (Egglegger)sesonghare;	<input type="radio"/> ✓	<input type="radio"/>
Egglegger h9 = skogshare;	<input type="radio"/>	<input type="radio"/> ✓
Påskehare h10 = (Påskehare)skogshare;	<input type="radio"/>	<input type="radio"/> ✓

Maks poeng: 10