

# UNIVERSITETET I OSLO

## Det matematisk-naturvitenskapelige fakultet

Eksempel på eksamen i : INF1000 — Grunnkurs i objektorientert programmering  
Gjennomgås på forelesning: Torsdag 14. november 2013  
Oppgavesettet er på: 11 sider  
Vedlegg : Ingen  
Tillatte hjelpemidler : Alle trykte og skrevne

- For hver oppgave er angitt det maksimale antall poeng du kan få hvis du svarer helt riktig. Summen av poengene er 171. Oversatt i tidsbruk tilsvarer hvert poeng grovt regnet ett minutt, slik at 10 poeng tilsvarer 10 minutters arbeid, osv.
- Dersom du savner opplysninger i oppgaven, kan du selv legge dine egne forutsetninger til grunn og gjøre rimelige antagelser, så lenge de ikke bryter med oppgavens "ånd". Gjør i så fall rede for forutsetningene og antagelsene du gjør.

### Oppgave 1 (20 poeng)

Er disse programlinjene lovlige i Java?

JA	NEI	
<input type="checkbox"/>	<input type="checkbox"/>	<code>double x = 1/1/1;</code>
<input type="checkbox"/>	<input type="checkbox"/>	<code>String s = "3.141" + 2.718;</code>
<input type="checkbox"/>	<input type="checkbox"/>	<code>String s = 3.141 + "2.718";</code>
<input type="checkbox"/>	<input type="checkbox"/>	<code>double i++ = 0;</code>
<input type="checkbox"/>	<input type="checkbox"/>	<code>String double[] = {"a", "b", "c"};</code>
<input type="checkbox"/>	<input type="checkbox"/>	<code>double z = new double;</code>
<input type="checkbox"/>	<input type="checkbox"/>	<code>while (int x &lt; 10) {x = x + 1;}</code>
<input type="checkbox"/>	<input type="checkbox"/>	<code>String s = 10 + '%';</code>
<input type="checkbox"/>	<input type="checkbox"/>	<code>int[][] p = new int[4][];</code>
<input type="checkbox"/>	<input type="checkbox"/>	<code>int i = 0; System.out.print(i = i);</code>

### Oppgave 2 (16 poeng)

a) Hvor mange ganger blir "ABCD" skrevet ut her:

```
for (int i=0; i<15; i+=2) {  
    System.out.println("ABCD");  
}
```

Svar: ..... ganger

b) Hvor mange ganger blir "ABCD" skrevet ut her:

```
int i = 0;
while (++i * ++i < ++i) {
    System.out.println("ABCD");
}
```

Svar: ..... ganger

c) Hvor mange ganger blir "ABCD" skrevet ut her:

```
int i = 0;
while (i < 10) {
    System.out.println("ABCD");
    if (i > 0) {
        i = -i - 1;
    } else {
        i = -i + 1;
    }
}
```

Svar: ..... ganger

d) Hvor mange ganger blir "ABCD" skrevet ut her:

```
int k = 5;
for (int i = 0; k > 0; i++) {
    k = Math.max(0, k-i);
    System.out.println("ABCD");
}
```

Svar: ..... ganger

### Oppgave 3 (5 poeng)

Anta at følgende kodelinjer utføres:

```
double x = 3.14;
int m = (int) (x-5);
int k = m + m * m - m;
System.out.println("k = " + k);
```

Hva blir skrevet ut på skjermen?

Svar: .....

## Oppgave 4 (10 poeng)

Skriv ferdig metoden under, som skal finne alle negative verdier i  $x$  og returnere summen av disse. Hvis f.eks. arrayen  $x$  inneholder verdiene  $-3, 5, -18, 2, 18$  og  $-12$ , så skal svaret som returneres være  $-33$  (fordi  $-3-18-12 = -33$ ). Dersom det ikke er noen negative verdier i  $x$ , eller arrayen  $x$  har lengde  $0$ , eller  $x = \text{null}$ , skal metoden returnere verdien  $0$ .

Svar:

```
int finnSumNegative(int[] x) {
```

```
}
```

## Oppgave 5 (15 poeng)

Betrakt følgende metode:

```
int minfunksjon(int k, int x){
    int sum = 0;
    int verdi = 2;
    for (int i=1; i<=k; i++) {
        sum += verdi * x;
        verdi *= verdi;
    }
    return sum;
}
```

Hvilken verdi returneres fra metodekallet `minfunksjon(4,1)`?

Svar: .....

## Oppgave 6 (15 poeng)

Skriv et fullstendig Java-program som ber brukeren om å oppgi et positivt heltall og som deretter leser inn dette tallet og skriver ut med ord hvilke sifre tallet har. Hvis f.eks. brukeren oppgir tallet 1532 skal altså programmet skrive ut:

En Fem Tre To

Svar:

## Oppgave 7 (20 poeng)

Restauranten PizzaKing har samlet inn data over et helt år som viser hvor mange pizzaer de solgte hver av dagene dette året. Disse dataene ligger i en array `antallPizzaer` som har lengde 365. Du skal anta at 1. januar var en mandag dette året, slik at `antallPizzaer[0]` er antall pizzaer som ble solgt mandag 1. januar, `antallPizzaer[1]` er antall solgte pizzaer tirsdag 2. januar, osv. Skriv en metode `lagStatistikk` som kan kalles slik:

```
lagStatistikk(antallPizzaer);
```

og som da skriver ut på skjermen en passe overskrift og deretter hvor mange pizzaer det ble solgt i gjennomsnitt hver mandag, tirsdag, ....., søndag det aktuelle året. Hvis du synes det blir enklere kan du se bort fra årets siste dag, siden du da sitter igjen med data for 52 hele uker ( $364 = 7 * 52$ ). Utskriften kunne starte slik (tallene er naturligvis bare eksempler og vil avhenge av verdiene i `antallPizzaer`):

### Salgsstatistikk for PizzaKing

-----

**Mandag : 31.4 pizzaer solgt i gjennomsnitt**

**Tirsdag : 63.2 pizzaer solgt i gjennomsnitt**

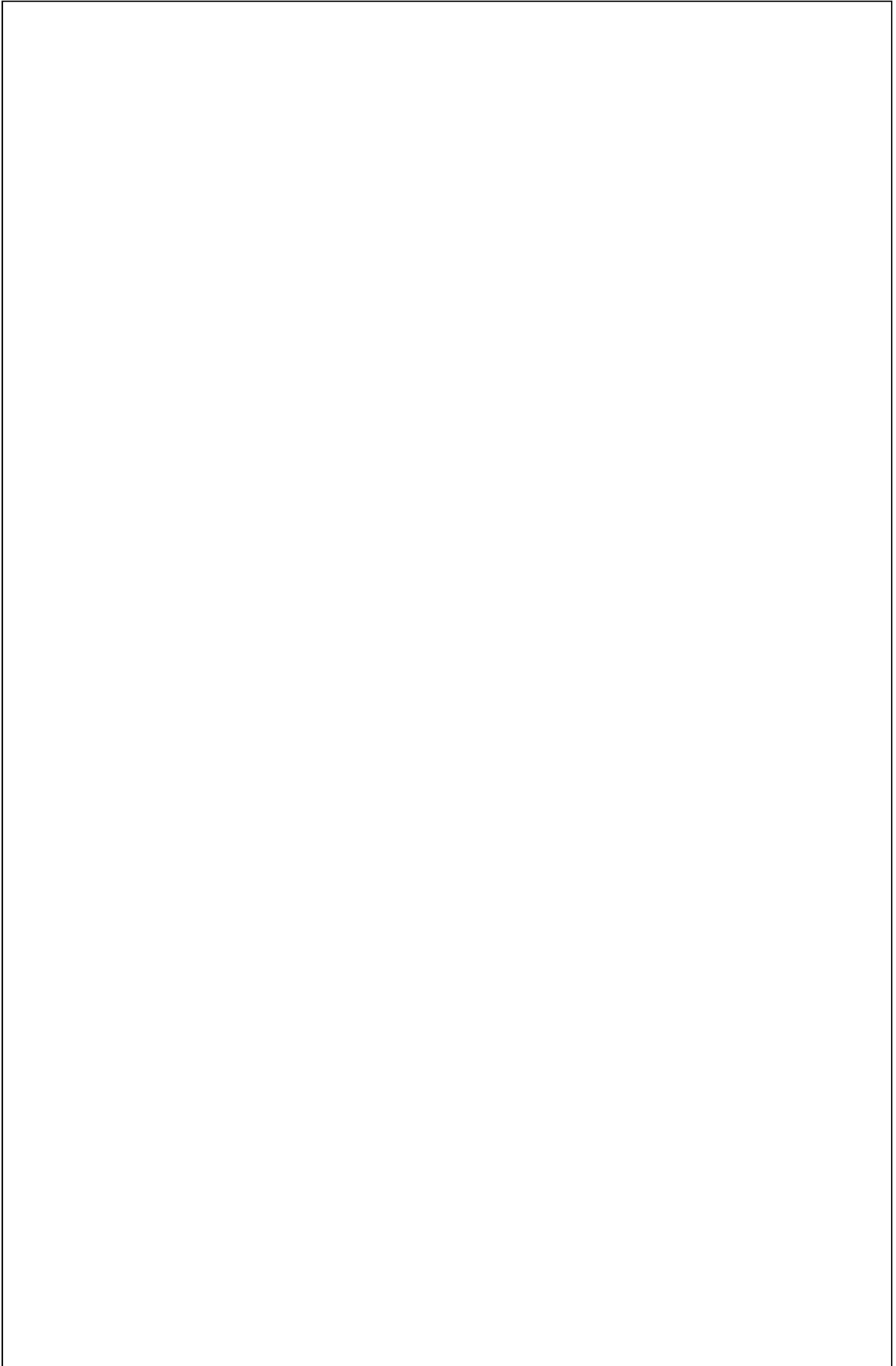
.....

.....

*Hint:* slå opp på s. 37 i læreboka hvis du vil bruke %-operatoren i denne metoden.

Svar:

*(Forts. neste side)*

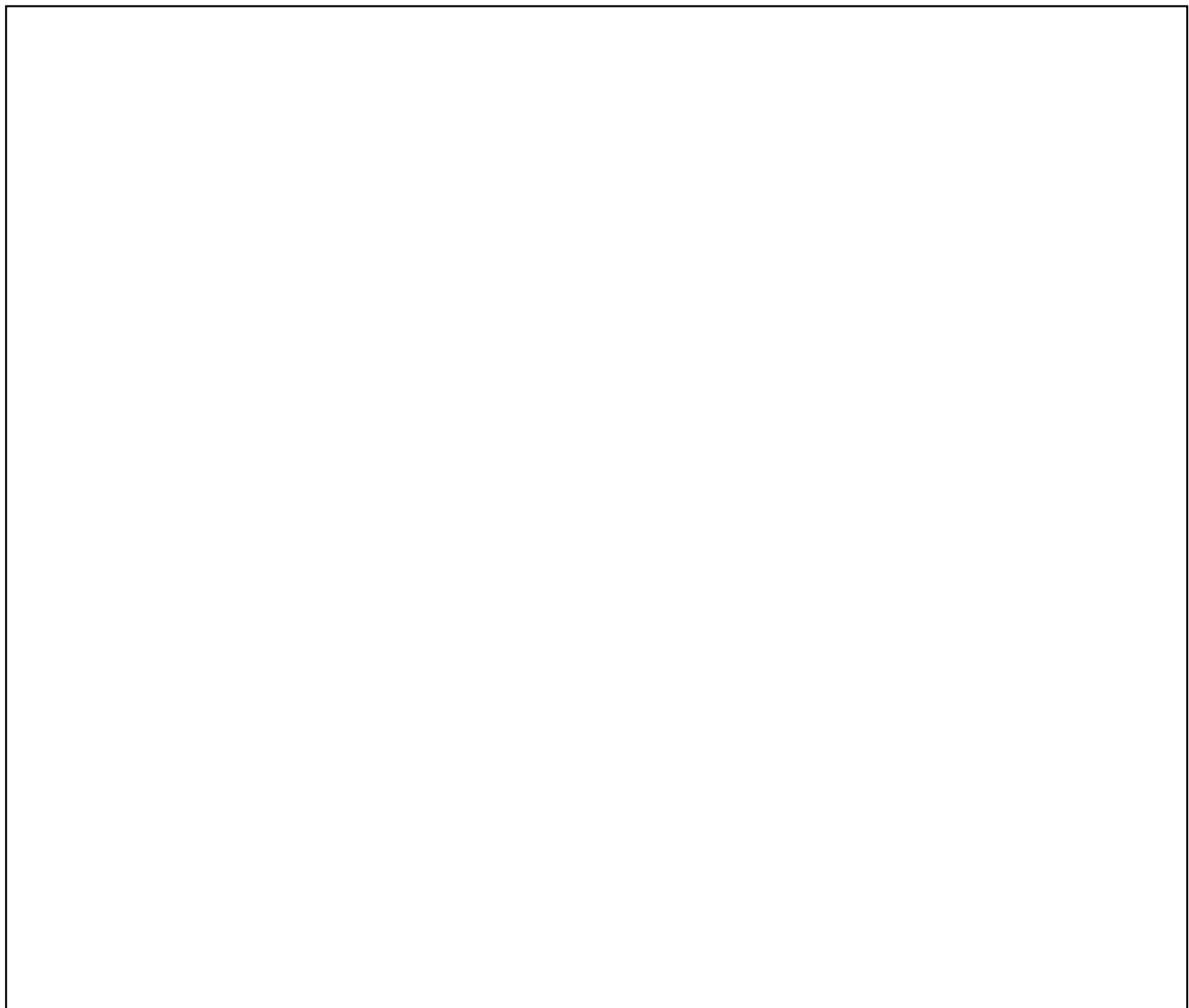


### Oppgave 8 (20 poeng)

Landet Utopia er i likhet med endel andre land (slik som USA) sammensatt av stater. Det er ialt 14 stater i Utopia, og hver av disse statene er igjen delt opp i et antall distrikter (antallet distrikter varierer fra 3 for de minste statene til 9 for den største staten). Hver stat har en statsregjering som består av alt fra 5 til 15 innbyggere fra dnne delstaten. En innbygger kan da ikke sitte i mer enn en statsregjering av gangen. Hvert distrikt har en distriktsregjering som består av 8 innbyggere. En innbygger i Utopia kan sitte i distriktsregjeringen for inntil tre distrikter på en gang (og muligens også i en statregjering). Hvert distrikt har en befolkning på inntil 1 million innbyggere. På grunn av uklare grenseopp ganger mellom distriktene kan en innbygger i Utopia være innbygger i inntil tre distrikter på en gang (at de da må betale dobbel eller trippel skatt og avtjene verneplikt i flere distrikter ser ikke ut til å bekymre dem – men dette er irrelevant for oppgaven). Hver stat i Utopia kan slutte en handelsallianse med så mange andre stater i Utopia de ønsker.

Tegn et UML-klassediagram med de følgende (Java-) klassene som kan brukes til å lage en UML-modell av Utopia: Land, Stat, Distrikt, Statsregjering, Distriktsregjering, Innbygger og Handelsallianse. Gi navn på relasjonene mellom disse klassene slik det er beskrevet ovenfor og plassér antall på begge sider av hvert forhold. Gjør egne forutsetninger ved behov, men forklar isåfall disse.

Svar:



## Oppgave 9 (30 poeng)

Under finner du skissen til et program som skal benyttes til å sensurere (erstatte) tvilsomme ord i en roman. Programmet skal først lese en liste med alle ordene som oppfattes som tvilsomme; disse ligger på filen **sensurord.txt** (du kan tenke deg at ordene i denne filen er atskilt med blanke tegn og/eller linjeskift). Deretter skal programmet lese filen **roman.txt** som inneholder teksten til en roman, sensurere alle tvilsomme ord og skrive resultatet til filen **sensurert-roman.txt**. Sensuren skal bestå i at tvilsomme ord i romanen erstattes med første bokstav i ordet som skal fjerne og at hver av de øvrige bokstavene i ordet som fjernes, erstattes med en stjerne. Du kan anta at det kun benyttes små bokstaver i teksten, slik at du slipper å tenke på å konvertere store bokstaver til små eller liknende. Hvis f.eks. romanen består av linjene

```
prøveeksamen stinker.  
jeg skulle gått på kino i stedet, eller tatt meg en ferie.
```

og ordene "stinker" og "ferie" ligger i listen av tvilsomme ord, så skal resultatet bli

```
prøveeksamen s*****.  
jeg skulle gått på kino i stedet, eller tatt meg en f*****.
```

Skriv ferdig programmet under ved å skrive metodene `lesUlovligeOrd(...)` og `sensurerTekst(...)` i klassen `Analyse`. Du må også deklareere nødvendig datastruktur i klassen `Analyse`. Du kan lagre sensurordene enten i en `HashMap` (med sensurordene som nøkler) eller i en `array`; det første gir et noe kortere og enklere program. Hint: du bør tenke nøye gjennom hvordan du skal håndtere spesialtegn i teksten (komma, punktum og linjeskift) og kommentere hvordan den sensurerte teksten eventuelt vil avvike fra originalteksten med hensyn til tilstedeværelse av slike tegn.

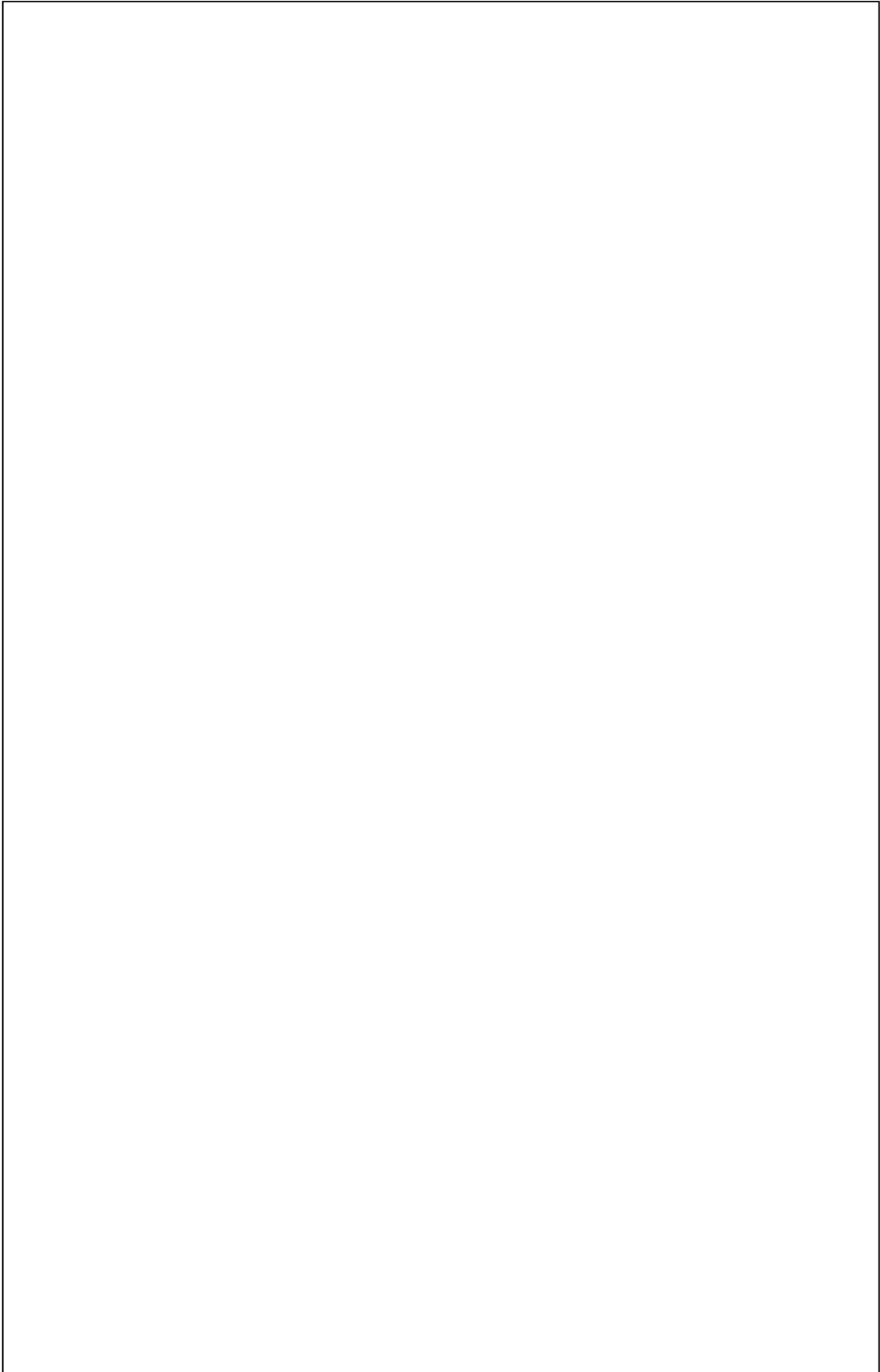
Svar:

```
import easyIO.*;  
import java.util.*;  
  
class Oppgave9 {  
    public static void main (String[] args) {  
        Analyse ana = new Analyse();  
        ana.lesUlovligeOrd("sensurord.txt");  
        ana.sensurerTekst("roman.txt", "sensurert-roman.txt");  
    }  
}  
  
class Analyse {
```

*(Forts. neste side)*



*(Forts. neste side)*



## Oppgave 10 (20 poeng)

Landet Uqbar har innført akkurat samme "Lov om behandling av personopplysninger" som i Norge og du skal bruke denne loven til å vurdere følgende:

På en bibliotek i Uqbar vil de lage et lite informasjonssystem om sine lånetagere, hvor de i tillegg til navn og adresse, vil lagre følgende opplysninger:

- Låntagernes alder, telefonnummer og e-post adresse.
- Hvilke bøker og videoer de har lånt de siste 5 årene.
- En tilhørende klassifikasjon av de bøkene og videoene som er lånt med kommentarer om hvilken type mennesker som pleier å låne disse bøkene.

Biblioteket har dårlig råd og håper å selge disse opplysningene til ulike firmaer som driver direkte-reklame via e-post. Ved at de ulike låntagerne kan klassifiseres etter hvilke bøker de leser, håper biblioteket å tjene relativt godt på dette opplegget.

Du skal nå drøfte om dette er greie opplysninger som biblioteket har rett til å samle inn eller om han må søke Datatilsynet i Uqbar om lov til dette. Begrunn svaret med å vise til konkrete paragrafer som du mener er relevante for dette registeret og de opplysningene biblioteket ønsker å samle inn, og hvorfor de ulike paragrafene du nevner er relevante.