



**Høgskolen i Telemark**

**EKSAMEN**

**I**

**5608 PROGRAMMERING**

**15.05. 2009**

Tid:	5 timar, 09:00 – 14:00
Målform:	Bokmål / Nynorsk
Sidetal:	5 med framsida
Hjelpemiddel:	Alt skriftleg materiale
Merknader:	Vekting er gitt ved starten av kvar oppgåve.
Vedlegg:	Ingen

Eksamensresultata blir offentleggjort på nettet, via Arena høgskole. Passordet til Arena har alle studentar fått i eige brev ved semesterstart. I tillegg finn ein lister med eksamensresultat på utsida av eksamenskontoret, men da treng ein kandidatnummeret. Noter deg difor kandidatnummeret ditt og ta vare på det.



**Oppgave 1**

**Vekting: 20%**

- 1 - a) Forklar kort hva som blir skrevet ut når programmet i Oppgave1a blir kjørt på en maskin.

```
public class Oppgave1a {
    public static void main(String[] args) {
        String a = "0123456789";
        String ap = a + ": ";
        for (int i = 0; i < a.length(); i++)
            if (i % 2 == 0)
                ap += a.charAt(i) + " ";
        System.out.println(ap);
    }
}
```

- 1 - b) Skriv et komplett java-program som ber om og leser inn ett ord, for så skrive ordet ut igjen baklengs i kun store bokstaver. F.eks. om "ord" leses inn, så blir utskriften "DRO".

**Oppgave 2**

**Vekting: 30%**

Karakterer fra diverse eksamener er lagret på sekvensielle tekstfiler. Filnavnet er satt sammen av eksamenssemester og emnekode, f.eks. "V09\_5002.txt". På hver av linjene på fila står det et studentnummer etterfulgt av tre bokstavkarakterer oppnådd på tre ulike deleksamener i det aktuelle emnet. Første deleksamen teller 20%, andre 30% og tredje 50% av den samlede emnekaraktene. Studentnummeret består av 6 siffer og de ulike dataene på fila er adskilt av ett blankt tegn (mellomrom). Se figuren nederst til venstre på sida.

Bokstavkarakterene omregnes slik: A = 5, B = 4, C = 3, D = 2, E = 1, F = 0. Til eksemplet på figuren under, får vi for studentnummer 071234 med karakterene A, B og C:

$$5*20/100 + 4*30/100 + 3*50/100 = (100 + 120 + 150)/100 = 370/100 = 3.7,$$

som avrundes til 4 og karakteren B (vanlig avrundingsregel opp til nærmeste heltall).

Skriv et komplett java-program som ber om og leser inn navnet på en slik fil (et aktuelt emne i et gitt eksamenssemester), og som for hvert studentnummer på denne fila beregner samlet emnekarakter. Programmet skal skrive resultatene inn på en ny fil med kun ett studentnummer og den tilhørende samlede emnekaraktene pr linje. La det nye filnavnet være "Kar\_" foran det opprinnelige filnavnet, f.eks. "Kar\_V09\_5002.txt".

V09_5002.txt	
071234	A B C
071362	A A B
072214	D B A
075123	F D E
:	
:	

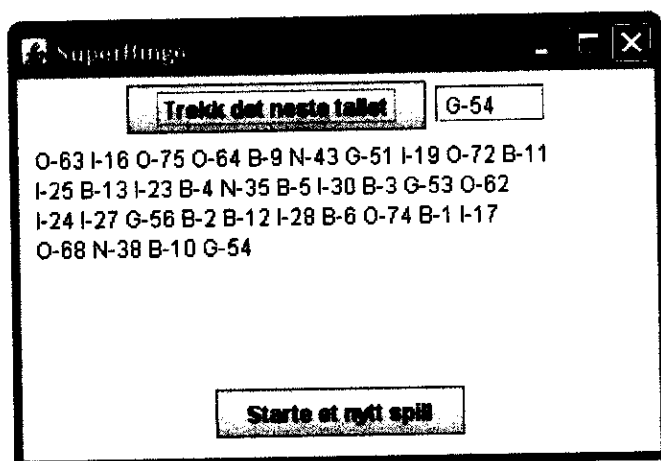
Kar_V09_5002.txt	
071234	B
071362	A
072214	B
075123	E
:	
:	

### Oppgave 3

Vekting: 30%

Bingo er et spill der det trekkes tilfeldig blant tallene 1-75, ett for ett, til en av spillerne har fått markert akkurat de 25 tallene vedkommende har på sitt spillebrett (eller helt til alle tallene er trukket ut). I Bingo er tallene inndelt i 5 grupper. Alle tallene i området 1-15 innledes med en B, alle 16-30 med en I, alle 31-45 med N, alle 46-60 med G og alle 61-75 med O. Trekkes f.eks. tallet 8, så angis dette som B-8, mens 54 altså angis som G-54.

Skriv et komplett java-program som basert på et grafisk brukergrensesnitt (GUI), kan brukes til trekking av alle tallene (ett for ett) i området 1-75. Tall med aktuell innledende bokstav vises i et eget tekstfelt, samtidig som det også legges til i kontrolloversikten over alle tall trukket ut så langt i spillet. Til kontrolloversikten brukes et tekstområde med 8 linjer og 30 kolonner, der det skrives maksimalt 10 tall pr linje. Det skal også være en knapp for å kunne starte et nytt spill.



(350 x 240 pix)

Eksempel på et Bingo-GUI

B	I	N	G	O
3	16	33	46	62
4	19	34	47	65
10	21	39	50	71
11	22	40	51	73
14	29	42	58	74

Eksempel på et Bingo-spillebrett.  
Trukne tall er markert med en ring.

### Oppgave 4

Vekting: 20%

Skriv klassen `Frimerke` som skal kunne brukes til å håndtere data i en frimerkesamling. Et `Frimerke`-objekt skal inneholde informasjon om: Land, årstall og påtrykt verdi. Klassen skal ha metoder for å kunne vise fram og endre aktuelle objektvariabler, og en `toString`-metode. Videre skal klassen ha en `compareTo`-metode slik at det er mulig å sortere `Frimerke`-objekt etter stigende årstall (de eldste frimerkene først).

Et program benytter `Frimerke`-klassen til å håndtere en (allerede registrert) frimerkesamling. Denne samlingen er ikke sortert på årstall. Skriv kun de instruksjonene som er nødvendige i et slikt program for å skrive ut alle de frimerkene i samlingen (ligger i tabellen `frimerkeliste`) som er utgitt i Norge, sortert etter stigende årstall.

**Oppgave 1**

**Vekting: 20%**

- 1 - c) Forklar kort kva som blir skrive ut når programmet i Oppgave1a blir kjørt på ein maskin.

```
public class Oppgave1a {
    public static void main(String[] args) {
        String a = "0123456789";
        String ap = a + ": ";
        for (int i = 0; i < a.length(); i++)
            if (i % 2 == 0)
                ap += a.charAt(i) + " ";
        System.out.println(ap);
    }
}
```

- 1 - d) Skriv eit komplett java-program som ber om og les inn eitt ord, for så skrive ordet ut igjen baklengs i kun store bokstavar. F.eks. om "ord" lesast inn, så blir utskrifta "DRO".

**Oppgave 2**

**Vekting: 30%**

Karakterar frå diverse eksamenar er lagra på sekvensielle tekstfiler. Filnamnet er sett saman av eksamenssemester og emnekode, f.eks. "V09\_5002.txt". På kvar av linene på fila står det eit studentnummer etterfølgt av tre bokstavkarakterar oppnådd på tre ulike deleksamenar i det aktuelle emnet. Første deleksamen tel 20%, andre 30% og tredje 50% av den samla emnekarakteren. Studentnummeret består av 6 siffer og dei ulike dataene på fila er adskilt av eitt blankt teikn (mellomrom). Sjå figuren nederst til venstre på sida.

Bokstavkarakterane omreknast slik: A = 5, B = 4, C = 3, D = 2, E = 1, F = 0. Til eksemplet på figuren under, får vi for studentnummer 071234 med karakterane A, B og C:

$$5*20/100 + 4*30/100 + 3*50/100 = (100 + 120 + 150)/100 = 370/100 = 3.7,$$

som avrundast til 4 og karakteren B (vanleg avrundingsregel opp til næraste heiltal).

Skriv eit komplett java-program som ber om og les inn namnet på ei slik fil (eit aktuelt emne i eit gitt eksamenssemester), og som for kvart studentnummer på denne fila reknar ut samla emnekarakter. Programmet skal skrive resultata inn på ei ny fil med kun eitt studentnummer og den tilhøyrande samla emnekarakteren pr line. La det nye filnamnet være "Kar\_" foran det opprinnelege filnamnet, f.eks. "Kar\_V09\_5002.txt".

V09_5002.txt	
071234	A B C
071362	A A B
072214	D B A
075123	F D E
:	
:	

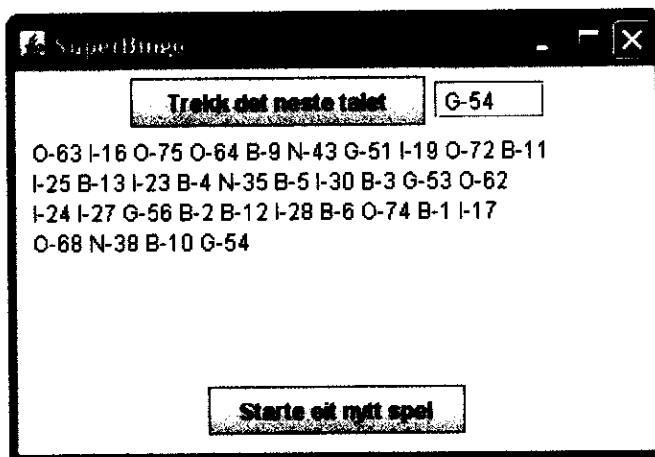
Kar_V09_5002.txt	
071234	B
071362	A
072214	B
075123	E
:	
:	

### Oppgave 3

Vekting: 30%

Bingo er eit spel der det trekkast tilfeldig blant tala 1-75, eitt for eitt, til ein av spelarane har fått markert akkurat dei 25 tala vedkommande har på sitt spillebrett (eller heilt til alle tala er trekt ut). I Bingo er tala inndelt i 5 grupper. Alle tala i området 1-15 innleiast med ein B, alle 16-30 med ein I, alle 31-45 med N, alle 46-60 med G og alle 61-75 med O. Trekkast f.eks. talet 8, så angis dette som B-8, mens 54 altså angis som G-54.

Skriv eit komplett java-program som basert på eit grafisk brukergrensesnitt (GUI), kan brukast til trekking av alle tala (eitt for eitt) i området 1-75. Tal med aktuell innleiande bokstav visast i eit eige tekstfelt, samtidig som det og leggast til i kontrolloversikta over alle tal trekt ut så langt i spelet. Til kontrolloversikta brukast eit tekstområde med 8 liner og 30 kolonner, der det skrivast maksimalt 10 tal pr line. Det skal og være ein knapp for å kunne starte eit nytt spel.



(350 x 240 pix)

Eksempel på eit Bingo-GUI

B	I	N	G	O
3	16	33	46	62
4	19	34	47	65
10	21	39	50	71
11	22	40	51	73
14	29	42	58	74

Eksempel på eit Bingo-spelebrett.  
Trekte tal er markert med ein ring.

### Oppgave 4

Vekting: 20%

Skriv klassen `Frimerke` som skal kunne brukast til å handtere data i ei frimerkesamling. Eit `Frimerke`-objekt skal innehalde informasjon om: Land, årstal og påtrykt verdi. Klassen skal ha metodar for å kunne vise fram og endre aktuelle objektvariablar, og ein `toString`-metode. Vidare skal klassen ha ein `compareTo`-metode slik at det er mulig å sortere `Frimerke`-objekt etter stigande årstal (dei eldste frimerka først).

Eit program nyttar `Frimerke`-klassen til å handtere ei (allereie registrert) frimerkesamling. Denne samlinga er ikkje sortert på årstal. Skriv kun dei instruksjonane som er nødvendige i eit slikt program for å skrive ut alle dei frimerka i samlinga (ligg i tabellen `frimerkeliste`) som er utgitt i Noreg, sortert etter stigande årstal.