Avdeling for allmenne fag

EKSAMEN

801 OBJEKTBASERT PROGRAMMERING

19.05.2003

Tid:	4 timar / 5 timar
Målform:	Bokmål/nynorsk
Sidetal:	7 med framsida
Hjelpemiddel:	Alt skriftleg materiale
Merknader:	Ingen
Vedlegg:	Ingen

Eksamensresultata blir offentliggjort på lister med kandidatnummer, lagt i hyllene ved eksamensoppslagstavla, eller du kan gå inn på Arena høgskole – da trenger du heller ikkje kandidatnummeret.

Bokmål

Oppgavesettet består av tre deler. Noen deloppgaver skal kun besvares av de som har fritak for midtprøven, og dermed får utvidet sluttprøve (5 timer). Disse deloppgavene er merket: "Skal kun besvares av de som har 5 timers sluttprøve".

Vekting av deloppgaver i forhold til endelig karakter i kurset går fram av følgende tabell:

Deloppgave	Vekting 4 timer	Vekting 5 timer
1-a	10 %	10 %
1-b	10 %	10 %
1-c	10 %	10 %
1-d	Skal ikke gjøres	15 %
2-a	25 %	25 %
2-b	Skal ikke gjøres	5 %
3	25 %	25 %
Midtprøve	20 %	Fritatt

Les oppgaveteksten nøye, og prøv å svare på så mange deloppgaver som mulig. Lykke til!

Oppgave 1

I denne oppgaven skal du programmere enkeltstående metoder. Deloppgavene er uavhengige av hverandre.

1-a)

Lag en metode som returnerer antall negative verdier i en heltallstabell.

```
public int antall( int[] tab )
```

Eksempel: Hvis *tab* inneholder verdiene 10, 7, -34, 0, 7, -3, 87 skal metoden returnere 2.

1-b)

Lag en metode som "fletter" to sorterte heltallstabeller.

```
public int[] flett( int[] tab1, int[] tab2 )
```

Returtabellen skal altså være sortert og inneholde samtlige elementer fra *tab1* og *tab2*.

1-c)

Lag en metode som leser inn og returnerer en desimalkarakter fra bruker.

```
public float lesKarakter()
```

En lovlig karakter har ett desimal og er en verdi mellom 1.0 og 6.0. Hvis brukeren skriver en ulovlig karakter skal metoden gjenta innlesingen helt til inndata er korrekt.

Innlesing skal gjøres ved hjelp av metoden *JOptionPane.showInputDialog*. Eksempel:

```
String kar = JOptionPane.showInputDialog( "Skriv karakter:" );
```

Effekten av denne setningen er at det blir vist et dialogvindu på skjermen med meldingen "Skriv karakter:". Vinduet inneholder en tekstboks der brukeren kan skrive inn tekst. Når brukeren trykker *enter* i tekstboksen blir teksten brukeren har skrevet inn returnert fra metoden *JOptionPane.showInputDialog*, og blir her tilordnet variabelen *kar*.

1-d) Skal kun besvares av de som har 5 timers sluttprøve

Lag en metode som tegner en kalender for en bestemt måned på skjermen.

```
public void tegnKalender( Graphics g, int år, int mnd,
                          int førsteUkedag )
```

Parameteren *førsteUkedag* forteller hvilken ukedag denne måneden begynner på, og er en verdi mellom 1 og 7, der 1 betyr mandag, 2 betyr tirsdag og så videre. For å forenkle kodingen kan du fritt bruke metoden *antallDager*, som returnerer antall dager i en bestemt måned:

```
public int antallDager( int år, int mnd )
```

Eksempel som viser kalender for mai 2003, der *førsteUkedag* vil være 4 for torsdag:

Mandag	Tirsdag	Onsdag	Torsdag	Fredag	Lørdag	Søndag
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Kalenderen skal tegnes i øvre venstre hjørne av skjermområdet som *g* representerer.

Oppgave 2

I denne oppgaven skal du lage deler av et program for en bedrift som utfører oppdrag knyttet til vedlikehold av veier og jernbanelinjer. Arbeidsoppgavene som inngår i et oppdrag krever en bestemt kompetanse, for eksempel sertifikat for spesielle kjøretøy, eller førstehjelpskurs. Informasjon om hva slags kompetanse hver ansatt har, og hvilke ansatte som er ledige på forskjellige dager er lagret i en samling tekstfiler. Det er spesifisert en modellklasse som i konstruktøren leser inn data fra filene og tilbyr to søkemetoder:

```
public class KompetanseBase extends Object
{
  // Deklarasjon av instansvariable

  public KompetanseBase() { … }
  public String[] ledige( int år, int mnd, int dag ) { … }
  public String[] harKompetanse( String kompetanse ) { … }
}
```

Metoden *ledige* returnerer en sortert tabell med navn på ansatte som er ledige den aktuelle dagen. Metoden *harKompetanse* returnerer en sortert tabell med navn på ansatte som har den aktuelle kompetansen.

2-a)

Lag brukergrensesnittet til et program for å finne hvilke ansatte som kan settes inn på et bestemt oppdrag. Brukeren skal kunne skrive inn en dato og en kompetanse. Programmet skal som respons vise en sortert liste med ansatte som tilfredsstiller søkekriteriet. Du velger selv om programmet skal lages som en applet, eller som en selvstendig applikasjon.

Merk: Du skal ikke programmere klassen *KompetanseBase*, men bruke metoder i klassen for å produsere utskriften av ansatte.

Brukergrensesnittet skal se ut som følger:

Dato:	19.05.2003	Kompetanse:	Førstehjelpskurs

Andersen, Per

Hansen, Lise

Larsen, Lars

Pettersen, Anders

2-b) Skal kun besvares av de som har 5 timers sluttprøve

Forklar hva slags datastruktur modellklassen *KompetanseBase* bør bygge opp. Beskriv eventuelle ekstra klasser som du mener det er naturlig å definere. Du skal ikke programmere metoder til disse klassene, men det skal gå fram hvilke instansvariable hver klasse bør ha.

Oppgave 3

Lag en selvstendig applikasjon som leser inn en tekstfil og skriver ut til skjermen antall forekomster av små bokstaver, store bokstaver og sifre.

Navn på tekstfilen skal sendes med som kommandolinjeparameter, og du kan anta at filen er lagret på samme katalog som programmet. Utdata skal skrives til "konsollet" med metoden *System.out.println*.

Tips: Det er hensiktsmessig å bruke noen hjelpemetoder for å hente ut enkelttegn og avgjøre om disse er bokstaver eller sifre. *String*-metoden *charAt* returnerer tegnet på en bestemt posisjon i en tekststreng. De statiske *Character*-metodene *isLowerCase*, *isUpperCase* og *isDigit* avgjør om et tegn henholdsvis er en liten bokstav, en stor bokstav, eller et siffer. Metodene håndterer norske tegn korrekt.

Eksempel på bruk av hjelpemetodene:

```
String s = "En tekst";
char c = s.charAt( 3 );                  // c får verdien 't'
boolean b = Character.isDigit( '5' );    // b får verdien true
b = Character.isDigit( 'a' );            // b får verdien false
b = Character.isLowerCase( 'g' );        // b får verdien true
b = Character.isLowerCase( 'G' );        // b får verdien false
b = Character.isUpperCase( 'Æ' );        // b får verdien true
b = Character.isUpperCase( '8' );        // b får verdien false
```

Eksempel på tekstfil:

```
Dette er en kort tekstfil.
Den består av 2 linjer.
```

Utskrift for eksempelfilen:

```
Antall små bokstaver: 36
Antall store bokstaver: 2
Antall sifre: 1
```

Slutt på oppgaveteksten

Nynorsk

Oppgåva inneheld tre delar. Nokre deloppgåver skal berre løysast av dei som har fritak for midtprøva, og av den grunn får ei utvida sluttprøve (5 timar). Desse deloppgåvene er merkte: "Skal berre løysast av dei som har 5 timars sluttprøve".

Vekting av deloppgåver i forhold til endeleg karakter i kurset går fram av følgjande tabell:

Deloppgåve	Vekting 4 timar	Vekting 5 timar
1-a	10 %	10 %
1-b	10 %	10 %
1-c	10 %	10 %
1-d	Skal ikkje gjerast	15 %
2-a	25 %	25 %
2-b	Skal ikkje gjerast	5 %
3	25 %	25 %
Midtprøve	20 %	Fritatt

Les oppgåva nøye, og freist å svare på så mange deloppgåver som mogeleg. Lukke til!

Oppgåve 1

I denne oppgåva skal du programmere enkeltståande metodar. Deloppgåvene er uavhengige av kvarandre.

1-a)

Lag ein metode som returnerer talet på negative verdiar i ein heiltalstabell.

```
public int antall( int[] tab )
```

Døme: Viss *tab* inneheld verdiane 10, 7, -34, 0, 7, -3, 87 skal metoden returnere 2.

1-b)

Lag ein metode som "flettar" to sorterte heiltalstabellar.

```
public int[] flett( int[] tab1, int[] tab2 )
```

Returtabellen skal altså vere sortert og innehalde alle elementa frå *tab1* og *tab2*.

1-c)

Lag ein metode som les inn og returnerer ein desimalkarakter frå brukar.

```
public float lesKarakter()
```

Ein lovlig karakter har eitt desimal og er ein verdi mellom 1.0 og 6.0. Viss brukaren skriv ein ulovleg karakter skal metoden ta oppatt lesinga heilt til inndata er korrekt.

Lesing skal gjerast ved hjelp av metoden *JOptionPane.showInputDialog*. Døme:

```
String kar = JOptionPane.showInputDialog( "Skriv karakter:" );
```

Effekten av denne setninga er at det blir synt fram eit dialogvindauge på skjermen med meldinga "Skriv karakter:". Vindauget inneheld ein tekstboks der brukaren kan skrive inn tekst. Når brukaren trykkjer *enter* i tekstboksen blir teksten brukaren har skrive inn returnert frå metoden *JOptionPane.showInputDialog*, og blir her tilordna variabelen *kar*.

1-d) Skal berre løysast av dei som har 5 timars sluttprøve

Lag ein metode som teiknar ein kalender for ein bestemt månad på skjermen.

```
public void tegnKalender( Graphics g, int år, int mnd,
                          int førsteUkedag )
```

Parameteren *førsteUkedag* fortel kva for vekedag denne månaden begynner på, og er ein verdi mellom 1 og 7, der 1 tyder måndag, 2 tyder tysdag og så vidare. For å forenkle kodinga kan du fritt nytte metoden *antallDager*, som returnerer talet på dagar i ein bestemt månad:

```
public int antallDager( int år, int mnd )
```

Døme som syner ein kalender for mai 2003, der *førsteUkedag* vil vere 4 for torsdag:

Måndag	Tysdag	Onsdag	Torsdag	Fredag	Laurdag	Søndag
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Kalenderen skal teiknast i øvre venstre hjørne av skjermområdet som *g* representerer.

Oppgåve 2

I denne oppgåva skal du lage delar av eit program for ein bedrift som utfører oppdrag knytt til vedlikehald av vegar og jernbaneliner. Arbeidsoppgåvene som inngår i eit oppdrag krev ein bestemt kompetanse, til dømes sertifikat for spesielle køyretøy, eller førstehjelpskurs. Informasjon om kva slags kompetanse kvar tilsett har, og kva for tilsette som er ledige på ulike dagar er lagra i ein samling tekstfiler. Det er spesifisert ei modellklasse som i konstruktøren les inn data frå filene og tilbyr to søkemetodar:

```
public class KompetanseBase extends Object
{
  // Deklarasjon av instansvariable

  public KompetanseBase() { … }
  public String[] ledige( int år, int mnd, int dag ) { … }
  public String[] harKompetanse( String kompetanse ) { … }
}
```

Metoden *ledige* returnerer ein sortert tabell med namn på tilsette som er ledige den aktuelle dagen. Metoden *harKompetanse* returnerer ein sortert tabell med namn på tilsette som har den aktuelle kompetansen.

2-a)

Lag brukargrensesnittet til eit program for å finne kva for tilsette som kan settast inn på eit bestemt oppdrag. Brukaren skal kunne skrive inn ein dato og ein kompetanse. Programmet skal som respons syne ei sortert liste med tilsette som tilfredsstiller søkekriteriet. Du vel sjølv om programmet skal lagast som ein applet, eller som ein sjølvstendig applikasjon.

Merk: Du skal ikkje programmere klassa *KompetanseBase*, men nytte metodar i klassa for å produsere utskrifta av tilsette.

Brukargrensesnittet skal sjå ut som følgjer:

Dato:	19.05.2003	Kompetanse:	Førstehjelpskurs

Andersen, Per

Hansen, Lise

Larsen, Lars

Pettersen, Anders

2-b) Skal berre løysast av dei som har 5 timars sluttprøve

Forklar kva slags datastruktur modellklassa *KompetanseBase* bør byggje opp. Beskriv eventuelle ekstra klasser som du meiner det er naturleg å definere. Du skal ikkje programmere metodar til desse klassene, men det skal gå fram kva for instansvariablar kvar klasse bør ha.

Oppgåve 3

Lag ein sjølvstendig applikasjon som les inn ein tekstfil og skriv ut til skjermen talet på førekomstar av små bokstavar, store bokstavar og siffer.

Namn på tekstfila skal sendas med som kommandolineparameter, og du kan gå ut frå at fila er lagra på same katalog som programmet. Programmet skal skrive utdata til "konsollet" med metoden *System.out.println*.

Tips: Det er fornuftig å nytte nokre hjelpemetodar for å hente ut enkeltteikn og avgjere om desse er bokstavar eller siffer. *String*-metoden *charAt* returnerer teiknet på ein bestemt posisjon i ein tekststreng. Dei statiske *Character*-metodane *isLowerCase*, *isUpperCase* og *isDigit* avgjer i rekkefølgje om eit teikn er ein liten bokstav, ein stor bokstav, eller eit siffer. Metodane handterer norske teikn korrekt.

Døme på korleis hjelpemetodane kan nyttast:

```
String s = "En tekst";
char c = s.charAt( 3 );                 // c får verdien 't'
boolean b = Character.isDigit( '5' );   // b får verdien true
b = Character.isDigit( 'a' );           // b får verdien false
b = Character.isLowerCase( 'g' );       // b får verdien true
b = Character.isLowerCase( 'G' );       // b får verdien false
b = Character.isUpperCase( 'Æ' );       // b får verdien true
b = Character.isUpperCase( '8' );       // b får verdien false
```

Eksempel på tekstfil:

```
Dette er en kort tekstfil.
Den består av 2 linjer.
```

Utskrift for eksempelfila:

```
Antall små bokstaver: 36
Antall store bokstaver: 2
Antall sifre: 1
```

Slutt på oppgåva