Høgskolen i Telemark

Avdeling for allmenne fag

EKSAMEN

I

852 OBJEKTORIENTERT PROGRAMMERING

10.06.2002

Tid:	5 timar (0900 – 1400)
Målform:	Nynorsk / bokmål
Sidetal:	3 + 3
Hjelpemiddel:	Alle trykte og skrevne
Merknader:	Ingen
Vedlegg:	Ingen

Eksamen 852 Objektorientert Programmering 10.06.2002 - Nynorsk

Oppgåvene er uavhengige av kvarandre i den meining at sjøl om du ikkje har løyst éi oppgåve, kan du løyse neste som om forrige var løyst. Disponér tida godt, slik at du får gjort noko på alle oppgåvene. Om ei oppgåve er uklår så lag dine eigne vilkår, og forklar desse.

Oppgåve 1

Når tømmer skal hentast ut av skogen med tømmerbilar, må lasset "målast" for at skogeigaren skal få rett betaling for lasset. Det er to måtar å måle tømmeret på, lassmåling og stokkmåling.

Lassmåling vil seie at eit heilt lass på tømmerbilen vert målt opp. Kor høgt er lasset, kor breitt, kor lange er stokkane i gjennomsnitt, osv. Eit lassmålt lass må innehalde berre stokkar av same treslag og av same kvalitetskategori. Eit lassmålt lass er difor kjenneteikna ved 1) treslag, 2) kvalitet, og 3) volumet av lasset, målt i kubikkmeter (m^3).

Ved stokkmåling vert kvar tømmerstokk i lasset oppmålt for seg: treslag, lengde, diameter, kvalitet. Sidan kvar stokk målast for seg, bestemmer vi her at eit slikt lass kan innehalde stokkar av ulike treslag og kvalitetar, sjølv om dette ikkje er heilt i samsvar med det verkelege liv.

Det skal utviklast eit datasystem som skal halde styr på det tømmeret som kjem inn til eit sagbruk. Alle stokkane i eit lass er frå same skogeigar eller leverandør.
Fylgjande kode er laga til no, og skal ikkje endrast, men det vert lagt inn meir seinare.

```
abstract class Lass {
  abstract public float getVolum();
}

class LassMålt extends Lass {
  protected float lengde, høgde, breidd;
  protected int treslag;
  protected int kvalitet;
  public float getVolum() { ... };
}

class StokkMålt extends Lass {
  protected Vector stokkar;
  public float getVolum() { ... };
}

class Stokk {
  protected float lengde;
  protected float diameter;
  protected int kvalitet;
  protected int treslag;
  public float getVolum() { ... }
}
```

1a

Skriv konstruktør i class LassMålt som initialiserar alle dei fem instansvariablane ved hjelp av parametre, konstruktør i class StokkMålt utan parametre, og konstruktør i class Stokk med fire parametre. Vektoren "stokkar" i class StokkMålt skal kunne innehalde alle stokkane i lasset.

1b

Skriv ferdig metoden getVolum i tre utgåver: i class Stokk, i class LassMålt, og i class StokkMålt. Når det gjeld lassmåling kan du gå ut i frå at 40% av volumet på eit tømmerlass er berre luft! Ved stokkmåling ser vi på ein stokk som ein sylinder, volumet er tverrsnitt*lengde, og tverrsnittet av ein sylinder er π*radius^2.

1c

Skriv ein metode i class StokkMålt med fylgjande start:

```
public void nyStokk(float nyLengde, float nyDiameter,
   int nyKvalitet, int nyTreslag) { ... }
```

- som opprettar ein stokk og legg den til lasset, dvs. inn i vektoren "stokkar".

1d

For at stokkane skal nyttast på beste vis, må ein ofte leite etter stokkar med visse eigenskapar. Lag ein metode i class Stokk:

```
public boolean erGodNok(float l, float d, int k, int t) { ... }
```

- som returnerar true dersom stokken er god nok, false elles. "God nok" vil seie at stokken er av same treslag som det spørrast etter (t), at stokken har same eller høgare kvalitet (k), at stokken har same eller større lengde (l), og at stokken har same eller større diameter (d).

1e

Lag ein metode i class StokkMålt:

```
public Stokk finn(float l, float d, int k, int t) { ... }
```

- som finn ein stokk i lasset som er god nok, og returnerar referanse til stokken. Returner null dersom ingen stokk passar.

Oppgåve 2

(Denne oppgåva er framhald av oppgåve 1, så du må fyrst lesa den.)
Ved måleplassen sit det ei og registrerer alle måleresultata på eit papirskjema, men dette skal no "over på data". Det skal kunne registrerast lassmålte lass, her skal registrerast: lengde, høgde, breidd, treslag og kvalitet. Det skal også kunne registrerast stokkmålte lass, for kvar stokk skal registrerast lengde, diameter, treslag og kvalitet. Det er snakk om 8 treslag og maksimalt 4 kvalitetskategoriar.

2a

Teikn eit framlegg til brukargrensesnitt for denne applikasjonen. La panela vere innramma, slik at strukturen vert tydeleg.

2b

Teikn komponenthierarkiet for brukargrensesnittet du har teikna i 2a. Vis tydeleg kva slags komponentar du nyttar, og kva for layoutmanager dei forskjellige panela skal ha.

2c

Drøft *kort* kvaliteten på det brukargrensesnittet du har laga.

Oppgåve 3

3a

Lag ei klasse Point3D som subklasse av class Point, slik at den representerer eit punkt i 3 dimensjonar. Den skal ha konstruktør med tre parametre, metodar "move" og "translate" med tre parametre, og reimplementert "equals" og "toString"-metode. Nytt metodane i Point der det er mogeleg.

3b

Lag ei klasse Dimension3D som subklasse av class Dimension, slik at den representerer breidd, høgde og djubde. Lag konstruktør utan parametre, med tre parametre, og kopi-konstruktør (med Dimension3D som parameter). Reimplementer equals og toString. Nytt metodane i Dimension der det er mogeleg.

Lykke til!

Eksamen 852 Objektorientert Programmering 10.06.2002 - Bokmål

Oppgavene er uavhengige av hverandre i den forstand at selv om du ikke har løst én oppgave, kan du løse neste som om forrige var løst. Disponér tiden godt, slik at du får gjort noe på alle oppgavene. Om en oppgave er uklar så lag dine egne forutsetninger, og forklar disse.

Oppgave 1

Når tømmer skal hentes ut av skogen med tømmerbiler, må lasset "måles" for at skogeieren skal få rett betaling for lasset. Det er to måter å måle tømmeret på, lassmåling og stokkmåling.

Lassmåling vil si at et helt lass på tømmerbilen blir målt opp. Hvor høyt er lasset, hvor bredt, hvor lange er stokkene i gjennomsnitt, osv. Et lassmålt lass må inneholde bare stokker av samme treslag og av samme kvalitetskategori. Et lassmålt lass er derfor kjennetegnet ved 1) treslag, 2) kvalitet, og 3) volumet av lasset, målt i kubikkmeter (m^3).

Ved stokkmåling blir hver tømmerstokk i lasset oppmålt for seg: treslag, lengde, diameter, kvalitet. Siden hver stokk måles for seg, bestemmer vi her at et slikt lass kan inneholde stokker av ulike treslag og kvaliteter, selv om dette ikke er helt i samsvar med det virkelige liv.

Det skal utvikles et datasystem som skal holde orden på det tømmeret som kommer inn til et sagbruk. Alle stokkene i et lass er fra samme skogeier eller leverandør.
Følgende kode er laget til nå, og skal ikke endres, men det blir lagt inn mere senere.

```
abstract class Lass {
  abstract public float getVolum();
}

class LassMålt extends Lass {
  protected float lengde, høyde, bredde;
  protected int treslag;
  protected int kvalitet;
  public float getVolum() { ... };
}

class StokkMålt extends Lass {
  protected Vector stokker;
  public float getVolum() { ... };
}

class Stokk {
  protected float lengde;
  protected float diameter;
  protected int kvalitet;
  protected int treslag;
  public float getVolum() { ... }
}
```

1a

Skriv konstruktør i class LassMålt som initialiserer alle de fem instansvariablene ved hjelp av parametre, konstruktør i class StokkMålt uten parametre, og konstruktør i class Stokk med fire parametre. Vektoren "stokker" i class StokkMålt skal kunne inneholde alle stokkene i lasset.

1b

Skriv ferdig metoden getVolum i tre utgaver: i class Stokk, i class LassMålt, og i class StokkMålt. Når det gjelder lassmåling kan du gå ut i fra at 40% av volumet på et tømmerlass er bare luft! Ved stokkmåling ser vi på en stokk som en sylinder, volumet er tverrsnitt*lengde, og tverrsnittet av en sylinder er π*radius^2.

1c

Skriv en metode i class StokkMålt med følgende start:

```
public void nyStokk(float nyLengde, float nyDiameter,
   int nyKvalitet, int nyTreslag) { ... }
```

- som oppretter en stokk og legger den til lasset, dvs. inn i vektoren "stokker".

1d

For at stokkene skal benyttes på beste vis, må man ofte lete etter stokker med visse egenskaper. Lag en metode i class Stokk:

```
public boolean erGodNok(float l, float d, int k, int t) { ... }
```

- som returnerar true dersom stokken er god nok, false ellers. "God nok" vil si at stokken er av samme treslag som det spørres etter (t), at stokken har samme eller høyere kvalitet (k), at stokken har samme eller større lengde (l), og at stokken har samme eller større diameter (d).

1e

Lag en metode i class StokkMålt:

```
public Stokk finn(float l, float d, int k, int t) { ... }
```

- som finner en stokk i lasset som er god nok, og returnerar referanse til stokken. Returner null dersom ingen stokk passer.

Oppgave 2

(Denne oppgaven er fortsettelse av oppgave 1, så du må først lese den.)
Ved måleplassen sitter det en og registrerer alle måleresultatene på et papirskjema, men dette skal nå "over på data". Det skal kunne registreres lassmålte lass, her skal registreres: lengde, høyde, bredde, treslag og kvalitet. Det skal også kunne registreres stokkmålte lass, for hver stokk skal registreres lengde, diameter, treslag og kvalitet. Det er snakk om 8 treslag og maksimalt 4 kvalitetskategorier.

2a

Tegn et forslag til brukergrensesnitt for denne applikasjonen. La panelene være innrammet, slik at strukturen blir tydelig.

2b

Tegn komponenthierarkiet for brukergrensesnittet du har tegnet i 2a. Vis tydelig hva slags komponenter du bruker, og hvilken layoutmanager de forskjellige panelene skal ha.

2c

Drøft *kort* kvaliteten på det brukergrensesnittet du har laget.

Oppgave 3

3a

Lag en klasse Point3D som subklasse av class Point, slik at den representerer et punkt i 3 dimensjoner. Den skal ha konstruktør med tre parametre, metoder "move" og "translate" med tre parametre, og reimplementert "equals" og "toString"-metode. Benytt metodene i Point der det er mulig.

3b

Lag en klasse Dimension3D som subklasse av class Dimension, slik at den representerer bredde, høyde og dybde. Lag konstruktør uten parametre, med tre parametre, og kopi-konstruktør (med Dimension3D som parameter). Reimplementer equals og toString. Benytt metodene i Dimension der det er mulig.

Lykke til!