

## Oblig 3 tips – litt mer tips enn i oppgaven

lørdag 3. okt 2009

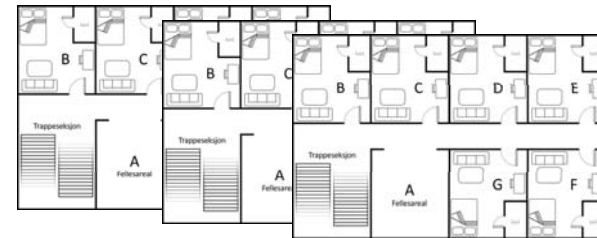
Arne Maus

## Obligatorisk oppgave 3 Gulbrand Grås husleiesystem

I denne oppgaven skal vi se på hans studenthus Utsyn. Utsyn består av 3 etasjer, nummerert fra 1-3. I hver etasje finner vi 6 hybler og et fellesareal. Hver hybel navngis med etasje + en bokstav. Første hybel har bokstaven B, siste hybel har bokstaven G. Fellesarealet er markert med bokstaven A. For eksempel 2A fellesarealet i andre etasje, mens 1E er hybel E i første etasje.

Gulbrand Grå krever 5000 kroner i måneden for en hybel de to nederste etasjene. Månedsprisen for en hybel i toppetasjen er 6000 kroner på grunn av utsikten.

Gulbrand har beregnet 1200 kroner i utgifter per hybel per måned (strøm, faste avgifter, avskrivninger), selv om det ikke bor noen der.



2

### Generell løsningsmetode

Husk vi skal lage en "kopi" av virkeligheten i Java-programmet

1. Finn **klassene** (hvilke objekter er det i problemet)
  1. Dataene som beskriver problemet (hvilke objekter har vi og hvor mange klasser er det ?)
2. Finn ut hvilke objekter som 'eier' objekter ( tegn **UML diagram**)
  1. Lag pekere (enkle pekere, arrayer eller HashMap'er) som binder systemet sammen
3. Finn ut **hva** systemet skal gjøre = metoder vi skal ha.
  1. Finn på et navn på hver av dem – foreløpig uten kode inni { }.
4. **Plasser metodene** (og andre datafelter) i de klassene de naturlig hører hjemme
5. Finn ut hvordan vi skal kjøre programmet – anta at vi har laget en klasse som er selve systemet – (class HusleieSystem)
  1. Lag et objekt av HusleieSystem (fra main)
  2. I konstruktoren leser du inn file(ne) i systemet og lager alle objektene som selve HusleieSystem 'eier' – for eksempel Et hybelhus 'Utsyn' med sine 21 hybler.
  3. Kall en 'kommandoløkke()' metode i HusleieSystem.
  4. Skriv ut nytt innhold av filer når 'kommandoløkke()' er ferdig
  5. Ferdig!
6. Lag et kjørende system med (bare) tomme metoder
7. Programmer hver metode for seg, test

3

### 1. Finn **klassene** - Substantiv-metoden : understrek alle

Gulbrand Grås firma HaiHus AS leier ut studenthybler i Ruritanias hovedstad Uqbar. Du skal hjelpe han å lage et system for å administrere utleie og effektivt kreve inn ubetalt husleie.

I denne oppgaven skal vi se på hans studenthus Utsyn. Utsyn består av 3 etasjer, nummerert fra 1-3. I hver etasje finner vi 6 hybler og et fellesareal. Hver hybel navngis med etasje + en bokstav. Første hybel har bokstaven B, siste hybel har bokstaven G. Fellesarealet er markert med bokstaven A. For eksempel 2A fellesarealet i andre etasje, mens 1E er hybel E i første etasje.

Gulbrand Grå krever 5000 kroner i måneden for en hybel de to nederste etasjene. Månedsprisen for en hybel i toppetasjen er 6000 kroner på grunn av utsikten.

.....

4

## 1. Finn **klassene** - Substantiv-metoden : understrek alle

Gulbrand Grås firma HaiHus AS leier ut studenthybler i Ruritanias hovedstad Uqbar. Du skal hjelpe han å lage et system for å administrere utleie og effektivt kreve inn ubetalt husleie.

I denne oppgaven skal vi se på hans studenthus Utsyn. Utsyn består av 3 etasjer, nummerert fra 1-3. I hver etasje finner vi 6 hybler og et fellesareal. Hver hybel navngis med etasje + en bokstav. Første hybel har bokstaven B, siste hybel har bokstaven G. Fellesarealet er markert med bokstaven A. For eksempel 2A fellesarealet i andre etasje, mens 1E er hybel E i første etasje.

Gulbrand Grå krever 5000 kroner i måneden for en hybel de to nederste etasjene. Månedsprisen for en hybel i toppetasjen er 6000 kroner på grunn av utsikten.

studenthybel, hovedstad, system, utleie, studenthus, etasje, hybel, månedspris, toppetasje, kroner, oppgave, etasje, fellesareal, bokstav, husleie,...

5

## 1. Substantivene er *forslag* til klasser

**Alt** for mange

### A. Noen **kutter vi ut**

- A. Angår ikke datasystemet: hovedstad
- B. Noen er egentlig handlinger (utleie, utdeling, innkreving) og blir kanskje metoder

### B. Noen blir til **enkle variable** inne i en klasse

- A. De som ikke inneholder noe mer, ingen indre struktur (etasjenr, bokstav, husleie, ..)

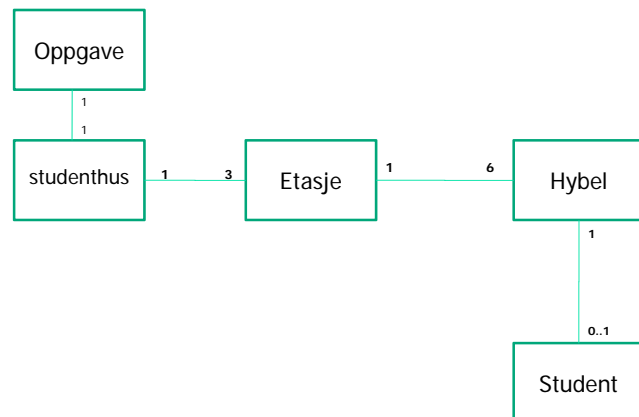
### C. Noen blir til **klasser**

- A. Typisk de som er gjentatt *mange ganger* (hybel, studenthus, etasje), men kanskje også bare én gang (student)

studenthybel, hovedstad, system, utleie, studenthus, etasje, hybel, månedspris, toppetasje, kroner, oppgave, etasje, fellesareal, bokstav, husleie,...

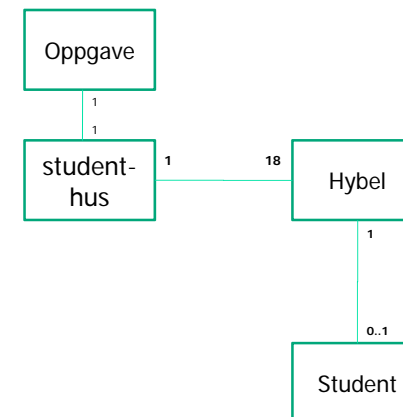
5

## 2. Tegn UML klassediagram (første forslag)2



7

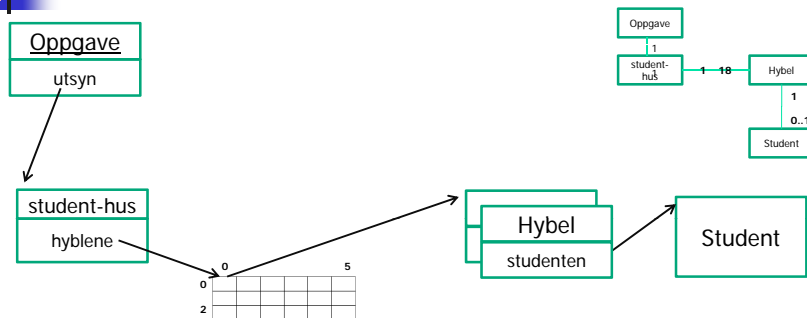
## 2. UML klassediagram (forslaget fra oppgave-teksten)



Hva er best ?

8

## 2b) Lag pekere ( nå henger det sammen)



9

## 3. Finn ut hva systemet skal gjøre = metoder vi skal ha

- Her er kravene til metoder fra oppgaven
  - alle (med ett mulig unntak) blir egne metoder
  - Tillegg: kommandoløkke() + meny()

0. Avslutt
1. Skriv oversikt
2. Registrer ny leietager
3. Registrer (frivillig) utflytting
4. Registrer betaling fra leietager
5. Månedskjøring av husleie
6. Sjekk om leietagere skal kastes ut
7. Lag backup av Hybeldata.txt

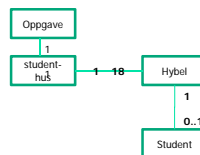
- Alle blir metoder – noen trenger parametre – det ordnes etterpå.

10

## 4. Plasser metodene (og andre datafelter) i de klassene de naturlig hører hjemme. Mange muligheter.

kommandoløkke() + meny()

0. Avslutt
1. Skriv oversikt
2. Registrer ny leietager
3. Registrer (frivillig) utflytting
4. Registrer betaling fra leietager
5. Månedskjøring av husleie
6. Sjekk om leietagere skal kastes ut
7. Lag backup av Hybeldata.txt



HybelHus: oversiktsTabell, innflyttingFraVenteliste, kommandoLøkke, meny, månedskjøring, sjekkOmUtkasting

Hybel: regInnFlytting, regUtflytting, regBetaling, sjekkOmUtkasting

Student:

11

## 4. Plasser datafelter i klasser

- Det som ikke ble klasser
  - etasjenr, bokstav, husleie,
- Student:
  - Navn, adresse,..?
- Andre variable
  - for eksempel når vi lager objekter (Student)

12

## 5. Kjøre systemet

```
class HusleieSystem {
    public static void main (String[] args) {
        StudentHus hh = new StudentHus ();
        hh.kommandoLøkke();
    }
}
```

13

## "hybeldata.txt" – systemets database

Først står det en linje for hver hybel, 21 linjer i alt, med hver linje på følgende form:

```
int etasje; char bokstav; String studentnavn; int saldo;
```

Et eksempel på en linje kan være:

```
2; C; Ole Brun; 2400;
```

For tomme hybler settes studentnavn lik "TOM HYBEL" og 0 i siste tall. Etter disse 18 linjene skal det stå en siste linje med format:

```
int måned; int år; int totaltAntallMåneder; int totalFortjeneste; 0;
```

14

## Lesing av redusert "hybeldata.txt":

```
2; C; Ole Brun; 2400;
3; A; TOM HYBEL; 3000;
```

Når vi har med en string 'sep' av separator tegn til en innlesningsmetode i klassen In, er det som leses en gruppe med tegn som er de tegnene som finnes mellom to skilletegnene i teksten sep

Før å si det enklere:

Først hoppes det over alle skilletegn i 'sep', så leses alt frem til neste skilletegn som det vi egentlig er interessert i.

Eks. `int i =fil.inInt(" ;");`

vil først hoppe over alle blanke og ; , så lese de neste tegnene til første 'blanke' eller ; og prøve å tolke det som et heltall.

15

```
1 import easyIO.*;
2
3 class Student {
4     String navn;
5     int saldo;
6     Student(String navn, int saldo) {
7         this.navn = navn;
8         this.saldo = saldo;
9     }
10    void skrivData() {
11        System.out.println("Student:" + navn + ", med saldo:" + saldo);
12    }
13 }
14
15 // 2; C; Ole Brun; 2400;
16 // 3; A; TOM HYBEL; 3000;
17
18 public class LesHybelData {
19     public static void main(String [] args) {
20         int i = 0;
21         int antLinjer = 2;
22
23
24         In hdata= new In("hybeldata.txt");
25
26         while (i++ < antLinjer) {
27             int et = hdata.inInt("; ");
28             char hbokstav = hdata.inChar("; ");
29             String navn = hdata.inWord("; ");
30             int saldo = hdata.inInt("; ");
31             Student s = new Student(navn,saldo);
32             s.skrivData();
33         }
34
35         hdata.close();
36     }
37 }
```

M:\INF1000\Oblig3tips>java LesHybelData  
Student:Ole Brun, med saldo:2400  
Student:TOM HYBEL, med saldo:3000

Merk ingen blanke skilletegn når vi leser inn navnet - HVORFOR

16

## Tabell

### 1. Skriv oversiktstabell:

Denne metoden skal skrive ut en oversikt som viser, for hver hybel: hybelnavn, studentnavn (eller "ledig" hvis hybelen er ledig), og saldo (eller skyldig beløp hvis det er i minus). Utskriften kan formateres i en tabell som vist under, eller på en annen måte.

```
Hybel Leietager          Saldo
-----
1A   ( ledig )          0
1B   Albert Johnsen     4300
1C   Johanne Ullersen   9000
1D   ( ledig )          0
     .... osv ....
```

Måned: 10.2008. Måneder i drift: 48. Totalfortjeneste: 274000.

Til slutt skrives det ut nåværende måned, antall måneder systemet har vært i drift, og total fortjeneste for Gulbrand.

17

## Tips 2

3. I skjelettet av løsningen ovenfor er det en Student-peker leietager i Hybel-klassen. For å teste om den peker på et objekt, kan man teste:

```
if (leietager == null)
```

`null` er et Java-ord som er pekerverdien 'intet objekt'. Dvs. if-testen slår til hvis `leietager` ikke peker på et objekt. `null` kan også brukes i tilordningssetninger for eksempel hvis `leietager` peker på et objekt, kan vi få fjernet det med: `leietager = null;`

18

## Tips 5

Hver gang metoden `tilkallTorpedo()` blir kalt skal du skrive til slutten av filen `torpedo.txt` (uten å overskrive det som allerede ligger der). Da må du åpne filen med `append`.

Det gjøres ved å ha en ekstra parameter når du åpner filen:

```
Out fil = new Out("Torpedo.txt", true);
```

Husk alltid å lukke filer når du er ferdig å lese/skrive til dem!

19

## Tips 3

4. Vi har behov for å plassere alle Hybel-objektene i en array. I program-eksempelet er det nyttig en todimensjonal array, hvor første indeksen går på etasjennummer og andre på hybel innen i en etasje. Nå er problemet at hybelen har et bokstavnavn (B,...,E), ikke et tall. Dette kan løses på følgende måte:

```
System.out.print("Gi etasje:");
int etg = tast.inInt();
System.out.print("Gi hybelbokstav:");
tast.skipWhite();
bokstav = tast.inChar();
i = (int) (bokstav - 'B');
```

Da vil: `hyblene[etg-1][i]` finne riktig peker i arrayen. Og motsvarende, hvis vi vet numeret 'i' til en hybel i etasjen, og vil ha bokstaven, gjør vi det slik:

```
char bokstav = (char) ('B' + i);
```

Begge disse omgjøringene fra char til heltall og motsatt, hviler på at bokstavene A,B,... ligger etter hverandre i kodingen av tegnsettet.

20

## Tips 4

```
if (new File("hybeldata.txt").exists() ) {
    // Kode for å lese fila:
    ... = new In("hybeldata.txt");
    // Les data og opprett objektene som beskrevet på fila...
    ...close();
} else {
    // Her kommer programmet hvis datafilen ikke finnes fra før.
    // Du må da opprette alle objektene som representerer
    // et hybelhus uten beboere
}
```

21

## Tips 5:

8. For å sikre at den månedlige beregningen av husleie og fortjeneste bare skjer en gang per måned, må du nok både spørre Gulbrand om hvilken måned han skal ha husleiekraft for, samt på fila "hybeldata.txt" ha opplysninger om hvilke måneder det allerede er skrevet ut husleie for.

22

```
import easyIO.*;

class Oblig3 {
    public static void main(String [] args) {
        Utsyn s = new Utsyn();
        s.menyLøkke();
    }
}

class Student {
    String navn; // studentens navn
    int saldo; // studentens saldo

    // evt. metoder for å behandle studentene
}

class Hybel {
    Student leietager; // peker på leietager. Er hybelen tom, er denne null
    boolean toppEtsasje; // true = 6000kr/mnd i leie
    // false = 5000kr/mnd i leie

    // evt. metoder for å behandle hyblene?
}
```

23

```
class Utsyn {
    Hybel[][] hyblene = new Hybel[3][6];

    // deklarer objektvariable for
    // økonomi-/månedsberegning

    In tast = new In();
    Out skjerm = new Out();
    String DATAFIL = "Hybeldata.txt";

    // Konstruktør for klassen Utsyn
    Utsyn() {
        // opprett hybler:
        // hyblene[etg][rom] = new Hybel();
        // Les inn fra datafil. Opprett studentobjekter
        // for de hyblene hvor det bor studenter:
        // hyblene[etg][rom].leietager =
        // new Student();
    }
}
```

```
void menyLøkke() {
    int kommando = -1;
    while (kommando != 0) {
        // skriv ut menyen
        skjerm.outln("Hovedmeny");
        skjerm.outln("0. Avslutt");

        // hent kommando fra bruker
        skjerm.out("Kommando: ");
        kommando = tast.inInt();

        // utfør kommando
        switch(kommando) {
            case 0: avslutt(); break;
            case 1: skrivOversikt(); break;
            case 2: regNyLeietager(); break;
            case 3: regFlytting(); break;
            case 4: regBetaling(); break;
            // ... fyll ut resten selv
            default: // gi feilmelding
        }
    }
}
```

24

```
// metoder for de forskjellige kommandoene i menyLøkke()
```

```
void skrivOversikt() { /* ... */ }  
void regNyLeietager() { /* ... */ }  
void regFlytting() { /* ... */ }  
void regBetalning() { /* ... */ }
```

```
// ... fyll ut med minst 4 ekstra metoder
```

```
void avslutt() { /* ... */ }  
}
```



Lykke til !