

Oblig3Hybelhus (INF1000 - Høst 2011) – én av to mulige oppgaver for oblig3 (du løser denne eller Oblig3Pi)

Gulbrand Grås Husleiesystem

Mål: Formålet med oppgaven er å gi erfaring med å løse et større programmeringsproblem ved hjelp av klasser og objekter, dessuten trening i filbehandling, arrayer, metoder, og brukerinteraksjon via terminal.

Krav til levering

Frist for innlevering er fredag 21. oktober kl. 16.00. Krav til innlevering generelt av obliger på ffi (MÅ LESES) finner du [her](#). Husk at to **ikke** kan levere felles (kopier av samme) løsning.

Det du skal leveres i Joly er én .java-fil (legges inn først) og filen hybeldata.txt. Programfilen skal hete Oblig3.java og inneholde alle klassene i besvarelsen - Java-nøkkelordet "package" skal ikke brukes. Filen hybeldata.txt er den som følger med oppgaven, men slik den ser ut etter at du har kjørt minst 5 innbetalinger av husleie og én månedskjøring (se oppgaven).

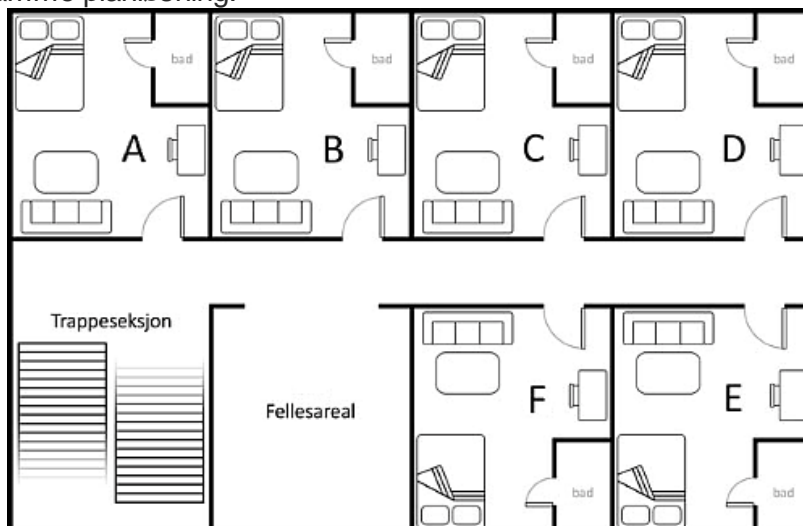
Krav til løsning – og noen tips

Øverst i programmet skal du kommentere besvarelsen, som hvilke antagelser du har gjort om uklarheter i oppgaveteksten, om noen deloppgaver mangler noe, og eventuelle spørsmål til gruppelærer. Selv om alt fungerer skal du skrive det i kommentaren.

Løsningen skal omfatte minst 3 klasser. Det er gode løsninger med 3 klasser (Oblig3, Hybelhus og Hybel), med 4 klasser (Oblig3, Hybelhus, Etasje og Hybel - eller Oblig3, Hybelhus, Hybel og Student) og med 5 klasser (Oblig3, Hybelhus, Etasje, Hybel og Student). Det finnes helt sikkert også andre gode løsninger; du må selv velge – og det er ikke sikkert at den med færrest antall klasser er enklest å lage. Flere tips finner du [her](#).

Oppgave

Gulbrand Grå har et hybelhus kalt Utsyn, med 18 studenthybler som han leier ut til studenter i Oslo. Du skal lage et system for å administrere utleie av hyblene i hybelhuset. Utsyn har 3 etasjer, nummerert fra 1 til 3. I hver etasje er det 6 hybler, kalt rom A til F, og et fellesrom. Hver hybel har et entydig «hybelnavn» som består av etasjenummer og rom-bokstav, f.eks. heter hybelen i rom C i andre etasje "2C". Nedenfor ser du en illustrasjon av en etasje. Alle etasjene har samme planløsning.



Økonomi

Inntekter: Gulbrand Grå leier ut hyblene i de to nederste etasjene for 6000 kroner i måneden, mens husleien for en hybel i toppetasjen er 7000 kroner på grunn av utsikten. Beboerne betaler husleie for sine hyblene til Gulbrand – og husleien dekker alle utgifter på hybelen unntatt strøm, som beboerne betaler utenom dette systemet.

Utgifter: Gulbrand har hyret inn firmaet Vedlikehold A/S som tar seg av alle løpende utgifter: Vedlikehold, reparasjoner, kommunale avgifter, nettabonnement i hyblene, og utstyr og strøm til fellesarealer. For dette betaler Gulbrand hver måned til Vedlikehold A/S: 1400 kr per hybel A-F i hver etasje, uansett om en hybel har beboer eller ikke, pluss 2100 kr per etasje for fellesarealer.

Datafilen "hybeldata.txt"

Når programmet starter (før hovedmenyen skrives ut på skjermen), skal programmet lese datafilen [hybeldata.txt](#). Her er det lagret blant annet informasjon om hvor lenge systemet har vært i drift og navn på de nåværende leietagerne. Første linje i filen inneholder fire heltallsverdier adskilt med semikolon: int måned; int år; int totalfortjeneste; int totaltAntallMåneder. Her er "int måned" og "int år" månedsnummeret og årstallet da månedskjøring sist ble utført, hvor måned er et tall i området 1 til 12, og årstallet er firesifret. Det tredje tallet er Gulbrands totale fortjeneste siden systemet ble satt i drift; og det siste tallet angir antall måneder systemet har vært i drift.

Deretter er det 18 linjer, en for hver hybel, med følgende format: int etasje; char bokstav; int saldo; String studentnavn. For tomme hybler skal studentnavnet lagres i datafilen som TOMHYBEL, med saldo 0. Du kan anta at alle studenter har unike navn. Alle disse dataene skal holdes oppdatert internt i programmet mens det kjører, og skal skrives tilbake til datafilen når programmet avsluttes. Slik kan Gulbrand starte og avslutte programmet uten å miste data mellom kjøringene.

Menyvalgene

Programmet skal være menystyrt. Det skal skrives ut på skjermen en meny over mulige ordre og be bruker om å taste en av disse. Hvis brukeren taster inn en ulovlig ordre, skal det gis feilmelding. Programmet skal gå i løkke og fortsette å lese og utføre ordre helt til brukeren taster ordre '7' for å avslutte. Bruker skal kunne gi 7 ordre:

1. Skriv oversikt
2. Registrer ny leietager
3. Registrer betaling fra leietager
4. Registrer frivillig utflytting
5. Månedskjøring av husleie
6. Kast ut leietagere
7. Avslutt

Detaljert beskrivelse av ordrene:

1. **Skriv oversikt**

På denne ordren skal programmet skrive ut en oversikt over alle hyblene, som viser

for hver hybel: hybelnavn, leietager-navn, og saldo. Dersom hybelen er ledig, skal teksten LEDIG skrives ut i stedet for leietager-navn, og saldoen vises som 0. Til slutt skal nåværende måned, år, antall måneder systemet har vært i drift, og totalfortjeneste skrives ut på skjermen.

Eksempel på hvordan oversikten kan se ut:

Hybel	Leietager	Saldo
1A	Ole Johan	6000
1B	Erik Smith	12000
1C	(LEDIG)	0
	...osv...	
Måned/år, og driftstid: 9/2010, 24 mnd. i drift		
Totalfortjeneste:		1400500 kr

2. Registrer ny leietager

Ordren brukes når en student ønsker å flytte inn og leie en av hyblene. Først sjekker programmet om det finnes ledige hybler, hvis ikke skriver det en melding om det og returnerer til hovedmenyen.

Hvis det finnes ledige hybler skal hybelnavnene til disse (f.eks 1C, 2B) skrives ut på skjermen, og så skal programmet spørre hvilken av disse studenten ønsker å leie. Bruker skal kunne taste inn ønsket **hybelnavn** som 1C, 2B, osv. Er valgt hybel ledig, skal programmet spørre om studentens navn, og registrere innflyttingen.

Studenten som flytter inn betaler samtidig et depositum på 10 000 kroner. Fra dette trekkes med én gang månedsleien for den første måneden og legges til totalfortjenesten (husk at månedsleien er avhengig av etasje). Det som er til overs blir studentens **saldo** (beholdningen i en slags forenklet bankkonto som leietagerne har hos Gulbrand). Når de betaler inn for husleie legges beløpet til i saldoen, og ved "månedskjøring" blir husleien trukket fra saldo. Saldoen vil være negativ hvis studenten skylder Gulbrand penger. Han trekker alltid én hel månedshusleie ved innflytting, uansett hvilken dag i måneden studenten flyttet inn.

Programmet skal til slutt skrive ut en beskjed på skjermen om at innflyttingen ble gjennomført. Beskjeden skal inneholde hybelnavnet (etasje+bokstav), studentens navn, og gjenværende saldo.

3. Registrer betaling fra leietager

Programmet skal spørre om et hybelnavn og beløpet som betales. Hvis hybelen som ble oppgitt ikke har beboer skal det skrives en feilmelding, ellers skal beløpet adderes til studentens saldo og en passende melding skrives på skjermen. Hver student må passe på å ha nok penger på sin saldo til å dekke husleien hver måned. Hvis saldo likevel var i minus etter forrige månedskjøring legges beløpet som var i minus til

totalfortjenesten til Gulbrand, mens resten av innbetalingen blir studentens nye saldo. Eks: Hvis saldoen var -1000 og studenten betalte 1500, så skal 1000 kr legges til totalfortjenesten og 500 legges til i studentens saldo.

4. **Registrer frivillig utflytting**

Programmet spør om *navnet på studenten* som ønsker å flytte ut, og leter deretter gjennom hyblene etter dette navnet. Hvis studenten ikke ble funnet, skal det gis en feilmelding, ellers skal det registreres i systemet at hybelen ikke lenger har beboer. Det forutsettes at leie er betalt for inneværende måned, og det utbetales ikke noe restbeløp selv om leietaker flytter før månedsslutt. Hvis saldoen er i minus skal programmet bare gi beskjed om at frivillig utflytting krever at husleie er betalt, og utflytting utføres ikke.

5. **Månedskjøring av husleie**

Gulbrand utfører denne ordren første dag i hver måned. Programmet ber brukeren bekrefte at det skal startes månedskjøring for måneden etter forrige månedskjøring. Hvis siste månedskjøring ble utført for måned 9 i år 2011, kan spørsmålet være "Ønsker du å utføre månedskjøring for måned 10/2011 (j/n)?" . Svarer brukeren n, returnerer programmet til hovedmenyen.

Svarer brukeren j, skal månedsnummer og evt. årstall oppdateres, og de månedlige overføringene utføres.

Programmet går gjennom alle hyblene: For hver hybel som har beboer trekkes månedsleien for hybelen fra studentens saldo og legges til Gulbrands månedsfortjeneste. Gulbrand belaster altså husleien forskuddsvis, for måneden som nettopp har begynt. Hvis noen leietagere ikke hadde nok i saldoen går denne i minus, men Gulbrands fortjeneste økes bare med det som var på saldo (han får inn resten av fortjenesten som forklart i menyvalg 3 og 6).

Gulbrands utgifter er det han betaler til Vedlikehold A/S, og dette trekkes fra månedsfortjenesten hans. Husk at det er forskjellige utgifter for hybler og fellesarealer.

Til slutt skal følgende skrives til skjerm:

- a. **Måned/år** som månedskjøringen gjelder for; **og driftstid** i antall måneder systemet har vært i drift, inkludert den nye måneden.
- b. **Månedens fortjeneste** er Gulbrands inntekter minus utgifter i denne månedskjøringen. Hvis du ønsker å vise andre inntekter fra innflyttinger eller annet siste måned, så skriver du dette ut som en egen post, den vanlige "månedens fortjeneste" skal kun vise regnskapet for månedskjøringen forklart ovenfor.
- c. **Totalfortjeneste** er Gulbrands nye totalfortjeneste, oppdatert med denne månedens fortjeneste.
- d. **Gjennomsnittlig månedsfortjeneste** regnes selvsagt ut som:
$$\text{totalfortjeneste} / \text{totaltAntallMåneder}.$$

Kast ut leietagere

Leietagere som skylder mer enn én husleie, blir kastet ut ved hjelp av torpedoen H. Hole når denne ordren utføres. Pengekravet til studenten som kastes ut er det han skylder i husleie pluss et utkastingsgebyr på 3000 kroner. Hole og Gulbrand deler gebyret likt. Gulbrands halvdel av gebyret pluss det studenten var i minus på saldoen legges til totalfortjenesten med én gang menyvalget kjøres (siden Hole alltid ordner disse sakene raskt), og hybelen registreres som ledig.

Programmet går gjennom alle hyblene og finner studentene med saldo lavere enn minus én månedsleie (husk de forskjellige leieprisene!). For hver av disse studentene skal du kalle følgende hjelpemetode (som du også skal programmere)

```
void tilkallHole(int etasje, int rom, int krav) { //...
```

Denne metoden skriver hybelnavn, studentnavn og pengekrav til skjerm og til filen **hole.txt**. Metoden skal ikke overskrive det som ligger på filen fra før, men legge til nye linjer på slutten (se evt. [tips](#)).

Avslutt

Ved utførelse av denne ordren skal nødvendige data skrives til hybeldata.txt: måned, år, totalfortjeneste, antall måneder i drift, samt leietager og saldo for alle hyblene. Deretter skal programmet avslutte.

Hjelpemetoder (frivillig)

Her er fire hjelpemetoder med inn- og ut-parametre som vil være nyttige hvis løsningen din både har klassen Student og klassen Hybel. Programmér disse (eventuelt med modifikasjoner som passer til datastrukturen du har valgt) og andre du får bruk for og benytt dem i programmet (se [tips](#) om hybelnavn-manipulasjon ved behov).

```
Hybel spørOmHybel(String ledetekst) {  
    // Skriver ut "ledetekst" på skjermen, Leser et hybelnavn fra  
    // tastatur, og returnerer den tilhørende Hybel-pekeren.  
}  
  
String finnHybelnavn(int etg, char rom) {  
    // Konverterer etg. og rom til et hybelnavn.  
}  
  
Student finnBeboer(String hybelnavn) {  
    // Finner og returnerer peker til leietageren som leier hybelen  
    // angitt i inn-parameteren "hybelnavn".  
}  
  
String stortTallTilString(int tall) {
```

```
// Lager en tekst med en mer lettlest form av inn-parameteren  
// "tall" med ett mellomrom for hvert 3. siffer (bakfra),  
// f.eks. 1400500 skal gi "1 400 500".
```

```
}
```

Hvis du har spørsmål, kommentarer, eller finner feil i oppgaveteksten kan du skrive disse i kurs-bloggen.

Lykke til!