

INF100/INF100-F - INNLEVERING 1 VÅREN 2006

Krav til innlevering

For at innleveringen skal godkjennes må følgende leveres:

- Oversikt

Et dokument som inneholder en oversikt over innleveringen. Den skal inneholde en kort beskrivelse av alle filene som blir levert inn og skal også oppgi navn, gruppenummer og epost-adresse. Hvis du ikke går på noen gruppe, skal du bruke gruppenummer 0.

- Kildekode

All kildekode skal selvsagt legges ved. Kildekoden skal være godt kommentert slik at det er lett å sette seg inn i programmenes virkemåte. Koden skal også være korrekt indentert (dvs. ha innrykk der programmet tilsier det) og lett å lese (f.eks. fornuftig bruk av variabelnavn). God indentering betyr generelt at all kode mellom to krøllparanteser ('{' og '}') indenteres et gitt antall 'mellomrom' i forhold til resten av koden. Det viktigste er å være konsistent og tenke på at indenteringen skal gjør koden lettere å lese.

- Kjøreeksempel

For hvert program skal det være med en fil som inneholder en logg av en typisk kjøring av programmet. Et slikt kjøreeksempel viser hvordan programmet brukes og hva programmet skriver ut. Under Linux kan filer med kjøreløgg lages slik:

1. Kjør kommandoen: `script logg.txt`
2. Kjør programmet ditt fra kommandolinjen v.h.a. kommandoen:
`java Mittprogram`
3. Avslutt med kommandoen `exit`

All skjermutskrift som ble produsert mens skriptet ble kjørt ligger nå på filen `logg.txt`.

Alle filene som skal leveres skal pakkes i en egen zip-fil med navn som er sammensatt av fornavnet ditt og gruppenummeret ditt. Dette gjøres slik: Hvis du f.eks. har fornavn Yngve og går på Gruppe 3, og du vil ha filene `Oppgave1.java`, `Oppgave2.java`, `oversikt.txt`, `logg1.txt` og `logg2.txt` med i din innlevering, lager du en zip-fil med navn `yngve3.zip` med kommandoen (alt på en linje):
`zip yngve3 Oppgave1.java Oppgave2.java oversikt.txt logg1.txt logg2.txt`

Du skal *ikke* levere inn filene som inneholder bytekode (filene med navn `*.class`). Bruk 0 som gruppenummer hvis du ikke går på noen gruppe.

Innleveringsvilkår

Innleveringsfristen er **mandag 13. februar 2006, Kl 16:00**. Besvarelsen skal bedømmes av gruppeleder, som setter en poengsum på en skala fra 0 (dårligste karakter) til

25 (beste karakter). Innlevering etter fristen eller ingen innlevering vil resultere i poengsummen 0.

Resultatet på de fire innleveringene må totalt være minst 40 for å få lov til å gå opp til eksamen. Resultatet utgjør da 20% av sluttkarakteren.

Det er tillatt at to studenter (ikke flere) samarbeider om innleveringen og leverer felles besvarelse. Besvarelsen pakkes da i en zip-fil som beskrevet over og begge fornavn brukes i navnet på filen, adskilt med 'X'. F.eks. 'yngveXove3.zip'.

Innlevering på internett

Innleveringene i INF100 skal leveres via internett. Alle studenter som har meldt seg på INF100 på studentportalen (<http://minside.uib.no>) vil få tilsendt brukernavn og passord. Studenter som ikke har meldt seg på faget må selv ta kontakt med Harald Barsnes (haraldb@ii.uib.no) snarest mulig for å få mulighet for innlevering via internett. (Brukernavn og passord er enda ikke utsendt, men vil bli sendt ut i god tid før innleveringsfristen.)

Praktisk informasjon

Innleveringer leveres på følgende nettside: http://nettkurs.uib.no/CMS_struts/upload.do. De forskjellige filene skal pakkes i et zip-arkiv, som beskrevet over. Det er zip-filen som skal leveres via internett. Når man har tastet inn brukernavn og passord, kan man velge filen man vil levere inn. Når filen er levert vil man få en kvittering, tilsvarende teksten under.

```
Filen er lastet opp.  
Brukernavn:haraldb  
Kurs:INF100  
Filnavn:harald3.zip  
Tid levert: Fri Feb 09 12:39:56 CEST 2006
```

Man kan levere flere ganger, det er den sist innleverte filen som blir registrert som innlevert. Dette er nyttig hvis man er uheldig og leverer feil fil. Ta kontakt med Harald Barsnes (haraldb@ii.uib.no) dersom det er noe som er uklart. Hvis du har glemt passordet ditt kan du få nytt tilsendt på http://nettkurs.uib.no/CMS_struts/forgotPassword.do. Du skal da oppgi e-post adressen du har registrert i studentportalen.

Oppgave 1a

Før det ble vanlig å bruke kalkulator i skolen var matematiske tabeller populære. I tillegg til gangetabellen kunne man der finne kvadratrøtter, logaritmer og mye annet nyttig. I denne oppgaven skal vi lage et program som skriver ut en liten matematisk tabell. Programmet skal lese inn minste og største heltall, og deretter skrive ut kvotient og rest ved heltallsdivisjon på 5 samt $n*n*n$ for alle heltall n mellom disse grensene (grensene inkludert). Merk at tabellen skrives ut slik at heltallene kommer i stigende rekkefølge.

Forslag til brukerdiallog (alle tall bak ':' er brukerens inndata, mens alt annet er skjermutskrift fra programmet):

Dette programmet skriver ut kvotienten og resten ved heltallsdivisjon på 5 samt $n*n*n$ for alle heltall n i et oppgitt intervall.

Gi minste heltall: 5
Gi største heltall: 18

n	n/5	rest	$n*n*n$
5	1	0	125
6	1	1	216
7	1	2	343
8	1	3	512
9	1	4	729
10	2	0	1000
11	2	1	1331
12	2	2	1728
13	2	3	2197
14	2	4	2744
15	3	0	3375
16	3	1	4096
17	3	2	4913
18	3	3	5832

Oppgave 1b

I denne oppgaven skal vi lage et program som lar brukeren spille på en enkel spille-automat, en såkalt enarmet banditt (engelsk: slot machine). Automaten fungerer slik:

1. Spilleren legger på et gitt antall kronestykker han/hun vil spille for.
2. Hvert spill koster 1 krone og det er ikke mulig å fylle på "mer penger underveis i en kjøring av programmet.
3. For hvert enkelt spill genererer automaten tre tilfeldige heltall mellom 1 og 3 (dvs. tallene 1, 2 og 3).
4. Dersom de tre tallene er like vinner spilleren en gevinst lik $2*\text{tallverdien}$ til de tilfeldige tallene. F.eks. vil resultatet | 3 | 3 | 3 | gi en gevinst på $2*3 = 6$ kroner, mens resultatet | 2 | 2 | 2 | gir en gevinst på $2*2 = 4$ kroner.
5. Automaten skal hele tiden holde greie på spillerens totale gevinst. Ved spillets slutt skal også spillerens totale fortjeneste (innsats - total gevinst) regnes ut.

Hint: Metoden `Math.random()` returnerer et tilfeldig flyttall i intervallet $[0, 1)$. Dvs. fra og med 0 og opp til (men ikke med) 1. Kodelinjen `tilfeldigTall = (int)(Math.random()*(n)+1)` vil dermed medføre at variabelen `tilfeldigTall` inneholder et tilfeldig tall mellom 1 og n .

Under er et forslag til hvordan skjermutskriften fra programmet kan se ut. Legg merke til at programmet gjør en liten pause i hver iterasjon, slik at spilleren får tid

til å se på resultatet av enkeltspillet. Neste spill kommer først etter input fra spilleren (trykker på 'Enter').

Spilleren bestemmer selv hvor mye penger han/hun vil putte på automaten og trykker deretter "Enter" for å starte spillet. Automaten genererer så tre tilfeldige heltall mellom 1 og 3. Spilleren får kun gevinst dersom alle de tre tallene er like. Gevinsten blir da lik tallverdien*2. Etter hvert enkeltspill får spilleren vite hvor mye han/hun har vunnet så langt. Ved spilllets slutt oppgis også den totale fortjenesten (innsats - gevinst).

Hvor mange kroner vil du putte på automaten?

3

Du puttet på 3 krone(r).

Trykk "Enter" for å starte det første spillet.

Resultat: |3|2|2|

Beklager ingen gevinst.

Total gevinst så langt: 0 kroner.

Trykk "Enter" for å starte det neste spillet.

Resultat: |2|2|2|

Du vant 4 kroner!

Total gevinst så langt: 4 kroner.

Trykk "Enter" for å starte det neste spillet.

Resultat: |3|3|1|

Beklager ingen gevinst.

Total gevinst så langt: 4 kroner.

Spillet er over.

Total gevinst: 4

Gratulerer du har en total fortjeneste på 1 krone(r)!