

DAT102 Oblig 1

Jakob Kallevik
Nick James Hipol

```
1 package no.hvl.data102.filmarkiv.klient;
2 import no.hvl.data102.filmarkiv.adt.FilmMarkivADT;
3 import no.hvl.data102.filmarkiv.impl.FilmMarkiv;
4
5 public class FilmMarkivMain {
6     public static void main(String[] args) {
7         FilmMarkivADT filma = new FilmMarkiv(100);
8         Meny meny = new Meny(filma);
9         meny.start();
10    }
11 }
12
```

```
<terminated> FilmMarkivMain [Java Application] /Library/Java/JavaVirtualMachines/jdk-20.jdk/Contents/Home/bin/java (28. jan. 2024, 23:30:41 - 23:31:18) [pid: 33787]

Film NR: 1
Filmskaper: ThisFilmskaper
Tittel: ATitle
År: 2000
Sjanger: ACTION
Filmselskap: ThisFilmselskap
-----
ATitle
HiTitle
MyTitle
ATitle
HiTitle
MyTitle
Skriv inn filnummer: 1
Skriv inn filmskaper: ThisFilmskaper
Skriv inn tittel: ATitle
Skriv inn år: 2000
Skriv inn sjanger: ACTION
Skriv inn filmselskap: ThisFilmselskap
-----
Film NR: 1
Filmskaper: ThisFilmskaper
Tittel: ATitle
År: 2000
```

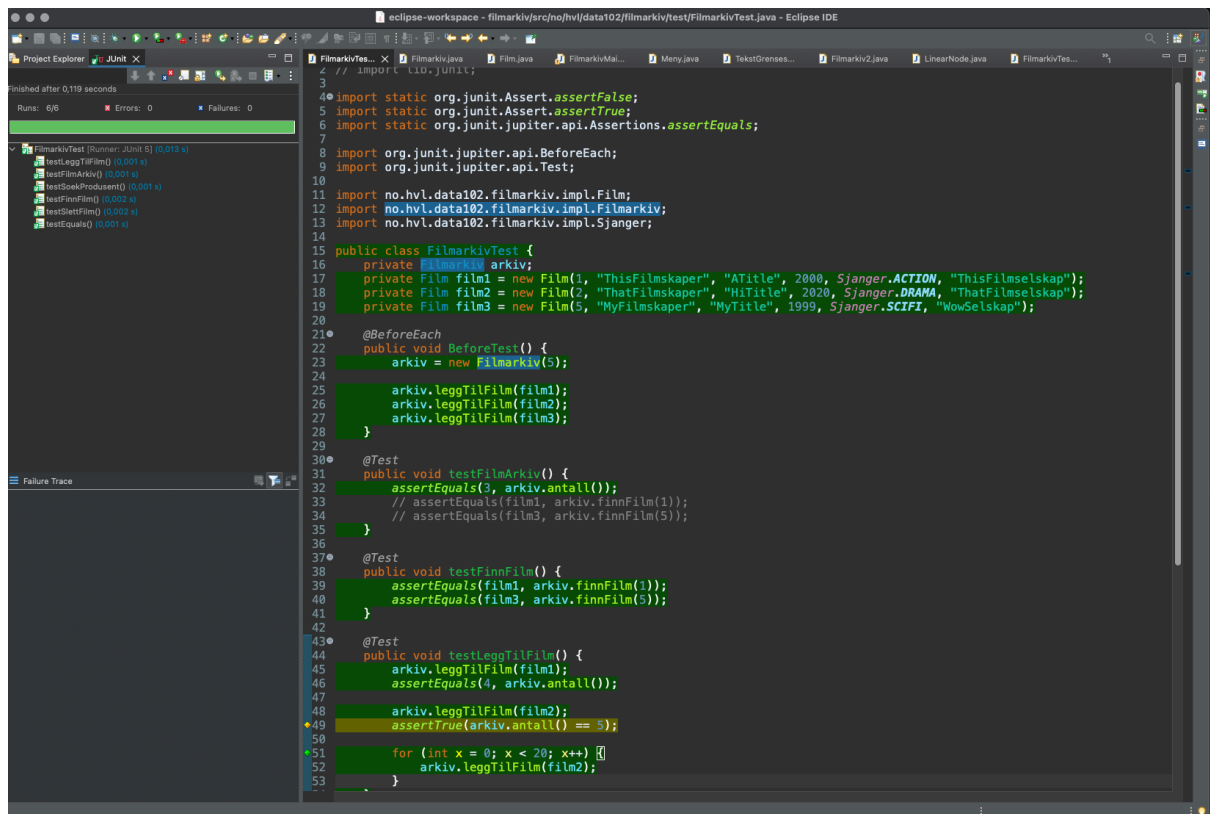
Programmet kjørt ved bruk av klassen Filmmarkiv

```
1 package no.hvl.data102.filmarkiv.klient;
2 import no.hvl.data102.filmarkiv.adt.FilmMarkivADT;
3 import no.hvl.data102.filmarkiv.impl.FilmMarkiv;
4 import no.hvl.data102.filmarkiv.impl.FilmMarkiv2;
5
6 public class FilmMarkivMain {
7     public static void main(String[] args) {
8         FilmMarkivADT filma = new FilmMarkiv2();
9         Meny meny = new Meny(filma);
10        meny.start();
11    }
12 }
13
```

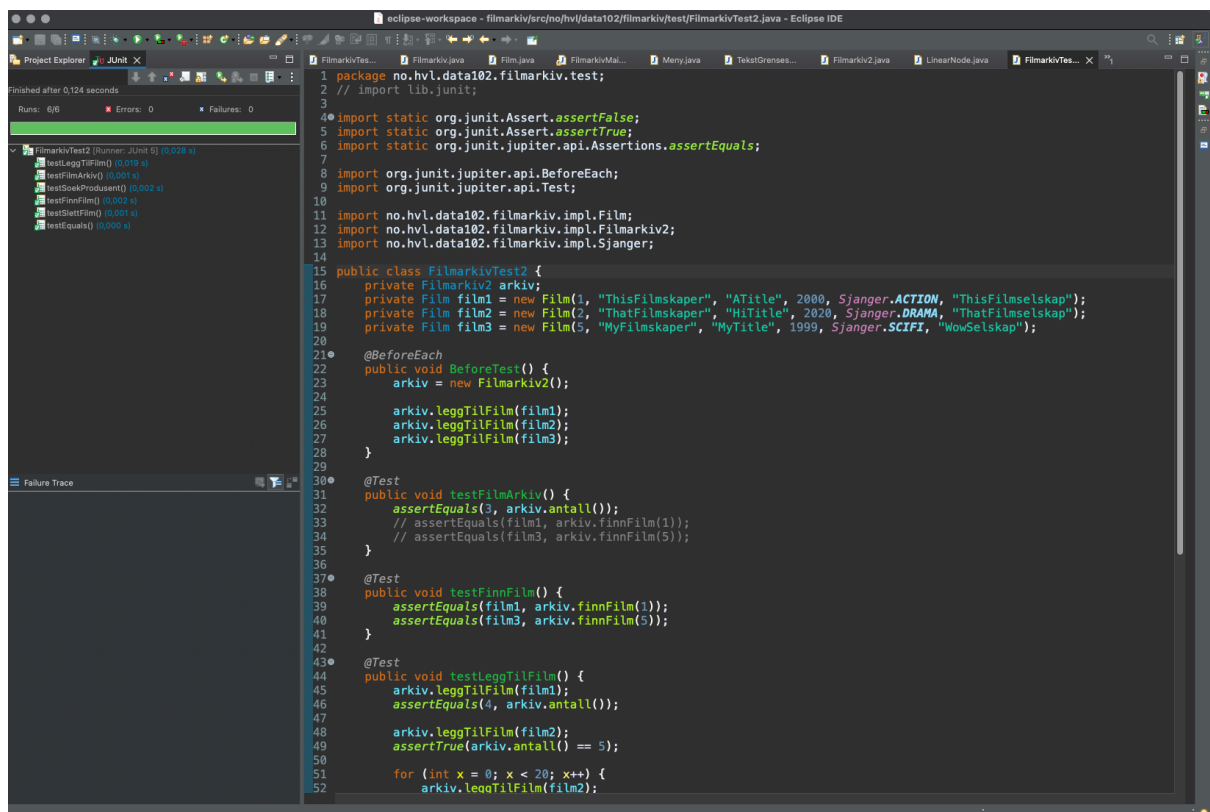
```
<terminated> FilmMarkivMain [Java Application] /Library/Java/JavaVirtualMachines/jdk-20.jdk/Contents/Home/bin/java (28. jan. 2024, 23:34:22 - 23:34:44) [pid: 33808]

MyTitle
ATitle
HiTitle
MyTitle
Skriv inn filnummer: 2
Skriv inn filmskaper: ThatFilmskaper
Skriv inn tittel: HiTitle
Skriv inn år: 2020
Skriv inn sjanger: DRAMA
Skriv inn filmselskap: ThatFilmselskap
-----
Film NR: 2
Filmskaper: ThatFilmskaper
Tittel: HiTitle
År: 2020
Sjanger: DRAMA
Filmselskap: ThatFilmselskap
-----
Antall filmer: 3
ACTION: 1
DRAMA: 1
HISTORY: 0
SCIFI: 1
```

Programmet kjørt ved bruk av klassen Firmarkiv2



JUnit for FilmmarkivTest av klassen Filmmarkiv



JUnit for FilmmarkivTest2 av klassen Filmmarkiv2

TEORI

a)

Størrelsesorden u/rykt i O-notasjon for algoritmen når veks;unksjonene er gi/ som:

i. $4n^2 + 50n - 10 \Rightarrow O(n^2)$

ii. $10n + 4\log(v2)n + 30 \Rightarrow O(n)$

iii. $13n^3 + 22n^2 + 50n + 20 \Rightarrow O(n^3)$

iv. $35 + 13\log(v2)n \Rightarrow O(\log n)$, siden 35 er en konstant term.

3 b) Gi/ algoritmen i oppgavebeskrivelse 3b). Finn antall Nlordninger (=) for algoritmen og effekNviteten u/rykt i O-notasjon.

Løkken halverer verdien av i for hver iterasjon, derfor er antall iterasjoner $\log(v2)n$, og antall Nlordninger blir: $1 + \log(v2)n * 1$.

EffekNviteten av algoritmen finner vi ved å Uerne konstan;aktoren 1, og da blir den dominerende termen $\log(v2)n$, og algoritmens effekNvitet er $O(\log n)$

O-notasjon:

3 c) Gi/ algoritmen i oppgavebeskrivelse 3c). Finn antall Nlordninger (=) for algoritmen og effekNviteten u/rykt i O-notasjon.

Antall iterasjoner i den ytre løkken er n, og antall iterasjoner i den indre løkken er $\log(v2)n$ fordi j dobles i hver iterasjon.

Dermed blir antall Nlordninger: $1 + n * \log(v2)n * 1$, og effekNviteten av algoritmen er $O(n * \log n)$

3 d) Vi ser på en sirkel med radius r. Da vil areal være gi/ med formelen $2 * \pi * r^2$ og omkrets med formelen $2 * \pi * r$. Angi i stor O-notasjon hvordan A og O vokser.

Areal: $O(r^2)$, Omkrets: $O(r)$. Det er altså fordi arealet og omkretsen NI sirkelen endres i forhold NI endringer i radien.

3 e) Finn antall sammenligninger i verste Nlfelle for algoritmen og effekNviteten u/rykt i O-notasjon.

Antall sammenligninger i verste Nlfelle for algoritmen og effekNviteten u/rykt i O-notasjon vil være summen av tallene fra n NI (n-1), fordi den ytre løkken går fra 0 NI n-2 og den indre løkken går fra indeks+1 NI n -1. Når antallet sammenligninger i den indre løkken kan beregnes som summen av tallene fra 1 NI (n-1) vet vi at vi har en aritmeNsk rekke. Det forteller oss at antall sammenligninger i verste Nlfelle er $(n(n-1))/2$.

For å finne effektivitet i O-notasjon Uerner vi konstant faktoren i u/rykket som og tar kun hensyn til den dominerende termen, som er $n \cdot n = n^2$.

Sammenfatning:

Antall sammenligninger i verste tilfelle: $(n(n-1)) / 2$. Effektivitet i O-notasjon: $O(n^2)$.

3 f) Hva er O-notasjonen for de ulike vekstfunksjonene under?

i. $t(v1)(n) = 8n + 4n^3 \Rightarrow O(n^3)$

ii. $t(v2)(n) = 10\log(v2)n + 20 \Rightarrow O(\log n)$

iii. $t(v3)(n) = 20n + 2n\log(v2)n + 11 \Rightarrow O(n)$

iv. $t(v4)(n) = 4\log(v2)n + 2n \Rightarrow O(n)$

Ranger de/effektivitet fra best til verst dersom n er stor:

1. $O(n^3)$
2. $O(n)$
3. $O(n)$
4. $O(\log n)$

3 g) Gi/ metoden i oppgavebeskrivelsen. Hvorfor er vekstfunksjonen til $T(n) = cn$, der c er en konstant.

Resultatet $T(n) = cn$ vil endres basert på verdien til n. c er en konstant, så det kommer an på verdien til n hva resultatet blir.

Hvordan stemmer resultatene med vekstfunksjonen?