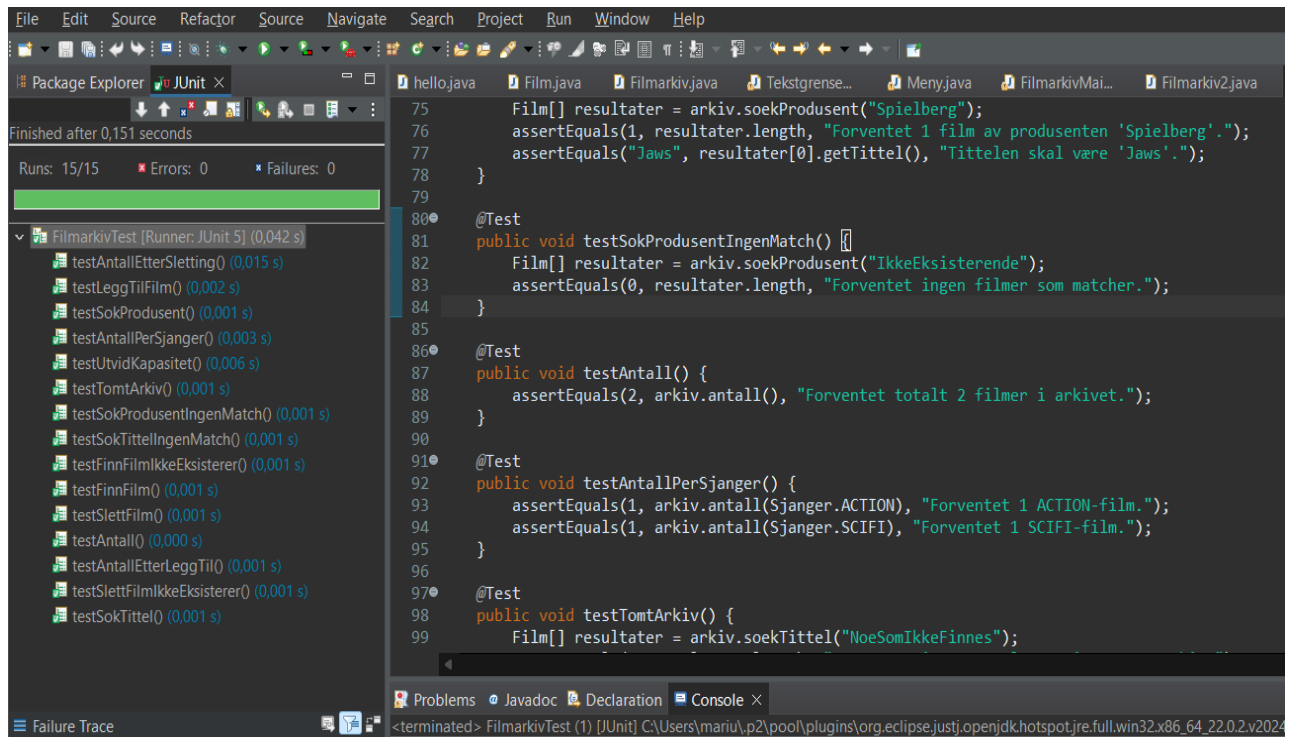
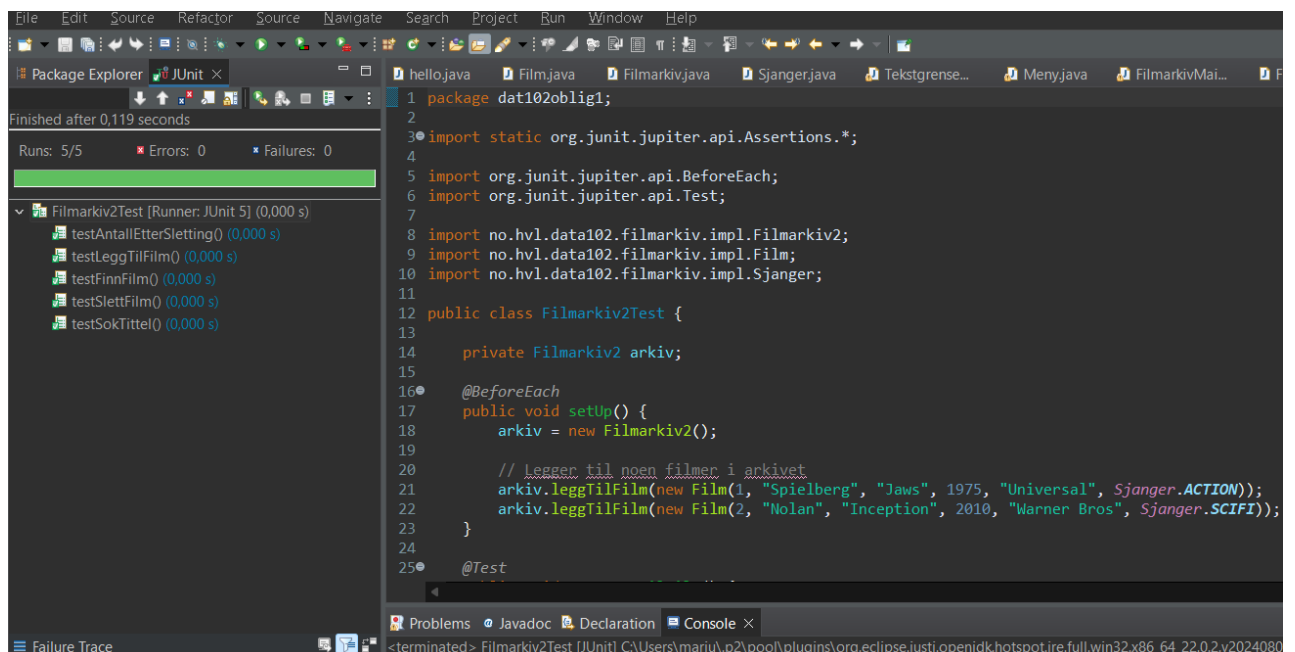


Gruppe 40: Endre Noren, Elias Daland, Kristoffer Albrigtsen og Marius L. Borgen



```
File Edit Source Refactor Source Navigate Search Project Run Window Help
Package Explorer JUnit X
Finished after 0,151 seconds
Runs: 15/15 Errors: 0 Failures: 0
FilmkivTest [Runner: JUnit 5] (0,042 s)
  testAntallEtterSletting() (0,015 s)
  testLeggTilFilm() (0,002 s)
  testSokProdusent() (0,001 s)
  testAntallPerSjanger() (0,003 s)
  testUtvidKapasitet() (0,006 s)
  testTomtArkiv() (0,001 s)
  testSokProdusentIngenMatch() (0,001 s)
  testSokTittelIngenMatch() (0,001 s)
  testFinnFilmIkkeEksisterer() (0,001 s)
  testFinnFilm() (0,001 s)
  testSlettFilm() (0,001 s)
  testAntall() (0,000 s)
  testAntallEtterLeggTil() (0,001 s)
  testSlettFilmIkkeEksisterer() (0,001 s)
  testSokTittel() (0,001 s)
hello.java Film.java Filmkiv.java Tekstgrense... Meny.java FilmkivMai... Filmkiv2.java
75 Film[] resultater = arkiv.sokProdusent("Spielberg");
76 assertEquals(1, resultater.length, "Forventet 1 film av produsenten 'Spielberg'.");
77 assertEquals("Jaws", resultater[0].getTittel(), "Littelen skal være 'Jaws'.");
78 }
79
80 @Test
81 public void testSokProdusentIngenMatch() {
82 Film[] resultater = arkiv.sokProdusent("IkkeEksisterende");
83 assertEquals(0, resultater.length, "Forventet ingen filmer som matcher.");
84 }
85
86 @Test
87 public void testAntall() {
88 assertEquals(2, arkiv.antall(), "Forventet totalt 2 filmer i arkivet.");
89 }
90
91 @Test
92 public void testAntallPerSjanger() {
93 assertEquals(1, arkiv.antall(Sjanger.ACTION), "Forventet 1 ACTION-film.");
94 assertEquals(1, arkiv.antall(Sjanger.SCIFI), "Forventet 1 SCIFI-film.");
95 }
96
97 @Test
98 public void testTomtArkiv() {
99 Film[] resultater = arkiv.sokTittel("NoeSomIkkeFinnes");
100 }
Problems Javadoc Declaration Console X
<terminated> FilmkivTest (1) [JUnit] C:\Users\mariu\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64.22.0.2.v2024
```



```
File Edit Source Refactor Source Navigate Search Project Run Window Help
Package Explorer JUnit X
Finished after 0,119 seconds
Runs: 5/5 Errors: 0 Failures: 0
Filmkiv2Test [Runner: JUnit 5] (0,000 s)
  testAntallEtterSletting() (0,000 s)
  testLeggTilFilm() (0,000 s)
  testFinnFilm() (0,000 s)
  testSlettFilm() (0,000 s)
  testSokTittel() (0,000 s)
hello.java Film.java Filmkiv.java Sjanger.java Tekstgrense... Meny.java FilmkivMai... Filmkiv2.java
1 package dat102oblig1;
2
3 import static org.junit.jupiter.api.Assertions.*;
4
5 import org.junit.jupiter.api.BeforeEach;
6 import org.junit.jupiter.api.Test;
7
8 import no.hvl.data102.filmkiv.impl.Filmkiv2;
9 import no.hvl.data102.filmkiv.impl.Film;
10 import no.hvl.data102.filmkiv.impl.Sjanger;
11
12 public class Filmkiv2Test {
13
14     private Filmkiv2 arkiv;
15
16     @BeforeEach
17     public void setUp() {
18         arkiv = new Filmkiv2();
19
20         // legger til noen filmer i arkivet
21         arkiv.leggTilFilm(new Film(1, "Spielberg", "Jaws", 1975, "Universal", Sjanger.ACTION));
22         arkiv.leggTilFilm(new Film(2, "Nolan", "Inception", 2010, "Warner Bros", Sjanger.SCIFI));
23     }
24
25     @Test
26     public void testAntallEtterSletting() {
27         assertEquals(2, arkiv.antall());
28         arkiv.slettFilm(1);
29         assertEquals(1, arkiv.antall());
30     }
31
32     @Test
33     public void testLeggTilFilm() {
34         arkiv.leggTilFilm(new Film(3, "Spielberg", "Jaws", 1975, "Universal", Sjanger.ACTION));
35         assertEquals(2, arkiv.antall());
36     }
37
38     @Test
39     public void testFinnFilm() {
40         Film film = arkiv.finnFilm(1);
41         assertEquals("Spielberg", film.getProdusent());
42         assertEquals("Jaws", film.getTittel());
43         assertEquals(1975, film.getAar());
44         assertEquals("Universal", film.getSjanger());
45     }
46
47     @Test
48     public void testSlettFilm() {
49         arkiv.slettFilm(1);
50         assertEquals(1, arkiv.antall());
51     }
52
53     @Test
54     public void testSokTittel() {
55         Film[] resultater = arkiv.sokTittel("Jaws");
56         assertEquals(1, resultater.length);
57         assertEquals("Spielberg", resultater[0].getProdusent());
58         assertEquals("Jaws", resultater[0].getTittel());
59         assertEquals(1975, resultater[0].getAar());
60         assertEquals("Universal", resultater[0].getSjanger());
61     }
62 }
Problems Javadoc Declaration Console X
<terminated> Filmkiv2Test [JUnit] C:\Users\mariu\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64.22.0.2.v2024080
```

Her er demo av hvordan programmet fungerer, kopiert fra konsoll. Fet tekst er input

Filmarkiv - Velg en operasjon:

- 1 - Legg til en ny film
- 2 - Finn en film etter nummer
- 3 - Slett en film
- 4 - Søk etter filmer etter tittel
- 5 - Søk etter filmer etter produsent
- 6 - Se antall filmer i en sjanger
- 7 - Vis alle filmer
- 8 - Avslutt programmet

Velg et alternativ (1-8): **1**

Hvor mange filmer vil du legge til? **1**

Legg til film nr. **1**

Skriv inn filmens nummer: **1**

Skriv inn filmens regissør: **Christofer Noland**

Skriv inn filmens tittel: **Interstellar**

Skriv inn produksjonsår: **2014**

Skriv inn filmens sjanger (1 - ACTION, 2 - DRAMA, 3 - HISTORY, 4 - SCIFI, 5 - SCIENCE_FICTION, 6 - FANTASY): **5**

Skriv inn filmens produsent: **Marius**

Filmarkiv - Velg en operasjon:

- 1 - Legg til en ny film
- 2 - Finn en film etter nummer
- 3 - Slett en film
- 4 - Søk etter filmer etter tittel
- 5 - Søk etter filmer etter produsent
- 6 - Se antall filmer i en sjanger
- 7 - Vis alle filmer
- 8 - Avslutt programmet

Velg et alternativ (1-8): **2**

Skriv inn filmnummeret du vil finne: **1**

Filmnummer: 1

Tittel: Interstellar

Regissør: Christofer Noland

År: 2014

Sjanger: SCIENCE_FICTION

Produsent: Christofer Noland

Filmarkiv - Velg en operasjon:

- 1 - Legg til en ny film
 - 2 - Finn en film etter nummer
 - 3 - Slett en film
 - 4 - Søk etter filmer etter tittel
 - 5 - Søk etter filmer etter produsent
 - 6 - Se antall filmer i en sjanger
 - 7 - Vis alle filmer
 - 8 - Avslutt programmet
- Velg et alternativ (1-8): **4**

Skriv inn tittel eller del av tittel du vil søke etter: **Christofer Noland**

Ingen filmer funnet med tittelen som inneholder: Christofer Noland

Filmarkiv - Velg en operasjon:

- 1 - Legg til en ny film
 - 2 - Finn en film etter nummer
 - 3 - Slett en film
 - 4 - Søk etter filmer etter tittel
 - 5 - Søk etter filmer etter produsent
 - 6 - Se antall filmer i en sjanger
 - 7 - Vis alle filmer
 - 8 - Avslutt programmet
- Velg et alternativ (1-8): **4**

Skriv inn tittel eller del av tittel du vil søke etter: **Interstellar**

Filmnummer: 1

Tittel: Interstellar

Regissør: Christofer Noland

År: 2014

Sjanger: SCIENCE_FICTION

Produsent: Christofer Noland

Filmarkiv - Velg en operasjon:

1 - Legg til en ny film

2 - Finn en film etter nummer

3 - Slett en film

4 - Søk etter filmer etter tittel

5 - Søk etter filmer etter produsent

6 - Se antall filmer i en sjanger

7 - Vis alle filmer

8 - Avslutt programmet

Velg et alternativ (1-8): **5**

Skriv inn produsent eller del av produsent du vil søke etter: **Christofer Noland**

Filmnummer: 1

Tittel: Interstellar

Regissør: Christofer Noland

År: 2014

Sjanger: SCIENCE_FICTION

Produsent: Christofer Noland

Filmarkiv - Velg en operasjon:

- 1 - Legg til en ny film
- 2 - Finn en film etter nummer
- 3 - Slett en film
- 4 - Søk etter filmer etter tittel
- 5 - Søk etter filmer etter produsent
- 6 - Se antall filmer i en sjanger
- 7 - Vis alle filmer
- 8 - Avslutt programmet

Velg et alternativ (1-8): **7**

Filmnummer: 1

Tittel: Interstellar

Regissør: Christofer Noland

År: 2014

Sjanger: SCIENCE_FICTION

Produsent: Christofer Noland

Filmarkiv - Velg en operasjon:

- 1 - Legg til en ny film
- 2 - Finn en film etter nummer
- 3 - Slett en film
- 4 - Søk etter filmer etter tittel
- 5 - Søk etter filmer etter produsent
- 6 - Se antall filmer i en sjanger
- 7 - Vis alle filmer
- 8 - Avslutt programmet

Velg et alternativ (1-8): **3**

Skriv inn filmnummeret du vil slette: **10**

Fant ikke film med nummer: 10

Filmarkiv - Velg en operasjon:

- 1 - Legg til en ny film
 - 2 - Finn en film etter nummer
 - 3 - Slett en film
 - 4 - Søk etter filmer etter tittel
 - 5 - Søk etter filmer etter produsent
 - 6 - Se antall filmer i en sjanger
 - 7 - Vis alle filmer
 - 8 - Avslutt programmet
- Velg et alternativ (1-8): **3**

Skriv inn filmnummeret du vil slette: **1**

Film med nummer 1 ble slettet.

Filmarkiv - Velg en operasjon:

- 1 - Legg til en ny film
 - 2 - Finn en film etter nummer
 - 3 - Slett en film
 - 4 - Søk etter filmer etter tittel
 - 5 - Søk etter filmer etter produsent
 - 6 - Se antall filmer i en sjanger
 - 7 - Vis alle filmer
 - 8 - Avslutt programmet
- Velg et alternativ (1-8): **10**

Ugyldig valg, prøv igjen.

Filmarkiv - Velg en operasjon:

- 1 - Legg til en ny film
- 2 - Finn en film etter nummer
- 3 - Slett en film
- 4 - Søk etter filmer etter tittel
- 5 - Søk etter filmer etter produsent
- 6 - Se antall filmer i en sjanger

7 - Vis alle filmer

8 - Avslutt programmet

Velg et alternativ (1-8): **8**

Avslutter programmet...

Oppgave 3

- i. $4n^2 + 50n - 10 \rightarrow O(n^2)$
- ii. $10n + 4 \log_2 n + 30 \rightarrow O(n)$
- iii. $13n^3 + 22n^2 + 50n + 20 \rightarrow O(n^3)$
- iv. $35 + 13 \log_2 n \rightarrow O(\log(n))$

b)

- Vi ser at det er to tilordninger $sum = sum + i$ og $i = i / 2$, løkken kjører $\log_2(n)$ og siden det er to tilordninger blir det $2(\log_2(n))$, effektiv O notasjon er $O(\log(n))$.

c)

- Vi har en tilordninger i $sum += i * j$. Den ytre løkken kjører «n» ganger. Den indre løkken kjører $\log_2(n)$ ganger for hver iterasjon i den ytre løkken så O notasjonen blir $n(\log_2(n))$.

d)

- $2\pi r^2$ og $2\pi r$, Arealet vokser med $O(r^2)$ og Omkretsen vokser med $O(r)$.

e)

- Ytre løkken går fra 0 til $n-2$, så den kjører $n-1$ ganger.
- Indre løkke kjører for hvert element i den ytre løkken starter den indre løkken på indeks + 1 og går til $n-1$. For hver iterasjon av den ytre løkken kjører den indre løkken «n - indeks - 1» ganger

$(n-1) + (n-2) + \dots + 1$ som gjør O-notasjon til $O(n^2)$

f)

- i. $t1(n) = 8n + 4n^3 \rightarrow O(n^3)$
- ii. $t2(n) = 10 \log_2 n + 20 \rightarrow O(\log(n))$
- iii. $t3(n) = 20n + 2n \log_2 n + 11 \rightarrow O(n \log n)$
- iv. $t4(n) = 4 \log_2 n + 2n \rightarrow O(n)$

Vekstfunksjonene fra verst til best:

- ii. $O(\log(n))$
- iv. $O(n)$
- iii. $O(n \log n)$
- i. $O(n^3)$

g)

Tid(n) lineær, altså $O(n)$. Dette betyr at tiden det tar å kjøre metoden skal være proporsjonal med n . Hvis vi måler tiden for $n=10^7$, 10^8 , og 10^9 , skal vi se en økning i tiden som er nær proporsjonal med verdien av n .

Vi ser at vi får et annet resultat enn forventet, dette er etter å ha kjørt et par ganger. Det mye som kan påvirke tiden det tar å kjøre. F.eks. `currentTimeMilli`.

Tid for $n = 100000000$: 2 ms

Tid for $n = 1000000000$: 4 ms

Tid for $n = 10000000000$: 20 ms

Tid for $n = 100000000000$: 178 ms