

# IN2010 --- innlevering 0

**Frist fredag 6. september kl 23.59**

*Denne innleveringa er ikke obligatorisk, men de som leverer og får godkjent vil få litt mindre å gjøre på den neste obligatoriske oppgaven. For å få godkjent må man løse minst to av de 4 oppgavene nedenfor. Informasjon om hvordan levere kommer på semestersida.*

## Oppgave 1 – en ineffektiv algoritme

Implementer dette javaprogrammet som ble vist på første forelesning:

```
class Fibonacci {

    public static void main(String[] args){
        Fibonacci f = new Fibonacci();
        long argument = Integer.parseInt(args[0]);
        if (argument < 0) argument = 0;
        System.out.println("f(" + argument + ") = "
                           + f.fibonacci(argument));
    }

    long fibonacci(long n) {
        if (n <= 1) {
            return n;
        } else {
            return fibonacci(n - 1) + fibonacci(n - 2);
        }
    }
}
```

Kjør programmet for  $n = 41$  til  $n = 50$  og lag en tabell med kolonnene  $n$ ,  $\text{fibonacci}(n)$  og antall millisekunder beregninga tok for hver  $n$ . Forklar med ord (norsk eller engelsk) hvorfor tidsbruken blir som den blir.

*En måte å gjøre dette på er ved hjelp av kommandoen time i linux. Her et eksempel for  $n=41$ :*

```
linuxprompt> time java Fibonacci 41
```

*Det finnes også biblioteksrutiner i Java som kan legges inn i koden for å gjøre dette. Det er også mulig å ta tida med en stoppeklokke.*

Hvor mange primitive operasjoner (jf. læreboka 1.1.3) utfører denne algoritmen for  $n$  lik 1?  $n$  lik 2?  $n$  lik 3?

I tillegg til kode og tabell skal du også tegne og levere et plott (jf. figur 1.1 i læreboka) over kjøretida til algoritmen.

## **Oppgave 2 - pseudokode**

Oversett algoritmen 1.2 i læreboka til Java.

## **Oppgave 3 – finne delliste med størst sum (jf. lærebokas avsnitt 1.3)**

Gitt ei lenkeliste hvor verdiene i nodene er et heltall (negativt, null eller positivt). Skriv en metode som finner dellista som har den største summen. Metoden skal kun returnere summen. Eks.: Hvis nodene i lista har verdiene

(-2, -4, 3, -1, 5, 6, -7, -2, 4, -3, 2)

skal metoden returnere 13 som er summen av dellista (3, -1, 5, 6).

## **Oppgave 4 – lag en oppgave selv**

Lag en oppgave (programmering, tegning, teori, flervalg, ...) med løsningsforslag innen et IN2010-tema, jf. ukesoppgavene som er gitt eller en oppgave om rekursjon eller lenkelister. Oppgaven skal ikke være større enn at det tar maks 15 minutter å løse den.