

## Obligatorisk oppgave 2: Algoritmeanalyse

Til info ble oppgavene kjørt på en PC med Windows 11 i Microsoft Visual Studio. Jeg opplevde raskere kjøretid på MacOS på oppgave 1 og 2, og tregere på 3 og 4. Disse er ikke inkludert her.

### Oppgave 1B:

- I den superlinære algoritmen kom jeg fram til at n-verdien 500\_050\_000 (500 millioner, og 50 tusen) var det næreste jeg kom 10 sekunder.
- Resultatet ble alltid omkring 10025 ms.

### Oppgave 2B:

- I den kubiske algoritmen kom jeg til slutt fram til at n-verdien 2500 var det nærmeste jeg kom 10 sekunder.
- Resultatet ble alltid omkring 10700 ms.

### Oppgave 3B:

- I den eksponentielle algoritmen slet jeg veldig med å finne et troverdig og stabilt svar.
- Som de andre funksjonene, brukte jeg long i for-løkka. Det som derimot skjedde var at alle n-verdier over 62 fikk samme kjøretid, uansett om det var 62, eller 5000.
- Jeg endret derfor fra long til Java sin innebygde BigInteger. Her kom jeg fram til at 31 var det nærmeste jeg kunne komme 10 sekunder.
- Resultatet ble alltid omkring 14175 ms.

### Oppgave 4B:

- I den kombinatoriske algoritmen kom jeg fram til at verdien 15\_500\_000\_000 var det nærmeste jeg kom 10 sekunder.
- Resultatet ble alltid omkring 10347 ms.