

Morten Stavik Eggen  
Georg Villhelm Seip

## Oppgave 1

print av å kjøre med 40 og drep hver 3.

```
Print List:
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
Kill: 3
Kill: 6
Kill: 9
Kill: 12
Kill: 15
Kill: 18
Kill: 21
Kill: 24
Kill: 27
Kill: 30
Kill: 33
Kill: 36
Kill: 39
Kill: 2
Kill: 7
Kill: 11
Kill: 16
Kill: 20
Kill: 25
Kill: 29
Kill: 34
Kill: 38
Kill: 4
Kill: 10
Kill: 17
Kill: 23
Kill: 31
Kill: 37
Kill: 5
Kill: 14
Kill: 26
Kill: 35
Kill: 8
Kill: 22
Kill: 40
Kill: 19
Kill: 1
Kill: 32
Kill: 13
Return: 28
Survivor: 28
```

Josephus må stå på plass nr 28.

Vår kode har kjøretid på  $O(n^2)$  tror jeg. For vi kjører 2 gjennomkjøringer av en liste på  $n$  størrelse i hverandre. 1 traversering og en delete som traverserer. Det hadde vel blitt  $O(n)$  bare om vi hadde laget en dobbel connection, så den enkelt finner "forrige" når den skal kjøre delete.

Stort sett veldig fornøyd med denne koden, spesielt printList funksjonen. Følte meg veldig smart når jeg fant ut at jeg kan gjøre den do{} while() loopen for å traversere lista en gang. Hadde opptil flere dårligere løsninger før det hvor jeg måtte finne størrelsen og sende inn den.

Her er også en print med 10 soldater og drep hver 4.

```
Print List:
1 2 3 4 5 6 7 8 9 10
Kill: 4
Kill: 8
Kill: 2
Kill: 7
Kill: 3
Kill: 10
Kill: 9
Kill: 1
Kill: 6
Return: 5
Survivor: 5
```

Morten Stavik Eggen  
Georg Villhelm Seip

## Oppgave 2

Kjøring med oppgave 1 som input

```
[Running] cd "c:\Users\Morten\Desktop\School\Semester5\IDATT2101_DataAlgo\Ov  
Stack Size: 72  
{ } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { }  
Stack Size: 0  
Alle tegnene løste seg  
[Done] exited with code=0 in 1.301 seconds
```

fjerner noe

```
[Running] cd "c:\Users\Morten\Desktop\School\Semester5\IDATT2101_DataAlgo\Ovin  
Stack Size: 71  
{ } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { } { }  
Stack Size: 1  
{  
Det ble ikke fullstendig match, det er en error her
```

```
[Done] exited with code=0 in 1.25 seconds
```

da ble en { ikke matchet med noen.

Ble ikke spesielt fornøyd med hvordan denne koden ble, følte litt klumsete. Spesielt når vi sender og mottar fra void\*, å hente kun første tegnet på på adressen føles litt som et plaster på et problem som burde løses på en annen måte.