

## Algoritmer og datastrukturer øving 4

### Deloppgave 1

Analyse av tidskompleksiteten for de relevante delene av programmet:

1. settInnBakerst(int verdi) - metoden:

- Denne metoden inneholder en while-løkke som kjører i verste fall  $O(n)$  tid, fordi den går gjennom listen for å finne den siste noden.
- Tidskompleksitet:  $O(n)$

2. josephusProblem(int n) - metoden:

- Den ytre while-løkken kjører inntil det bare er ett element igjen i listen, så det vil kjøre omtrent  $n$  ganger i verste fall.
- For hver iterasjon av den ytre while-løkken, er det en indre for-løkke som kjører  $m-1$  ganger for å hoppe over  $m-1$  noder.
- Kombinasjonen av den ytre while-løkken og den indre for-løkken gir en kompleksitet på  $O(n \times m)$ .
- Tidskompleksitet:  $O(n \times m)$

3. utførJosephus(int n, int k) - metoden:

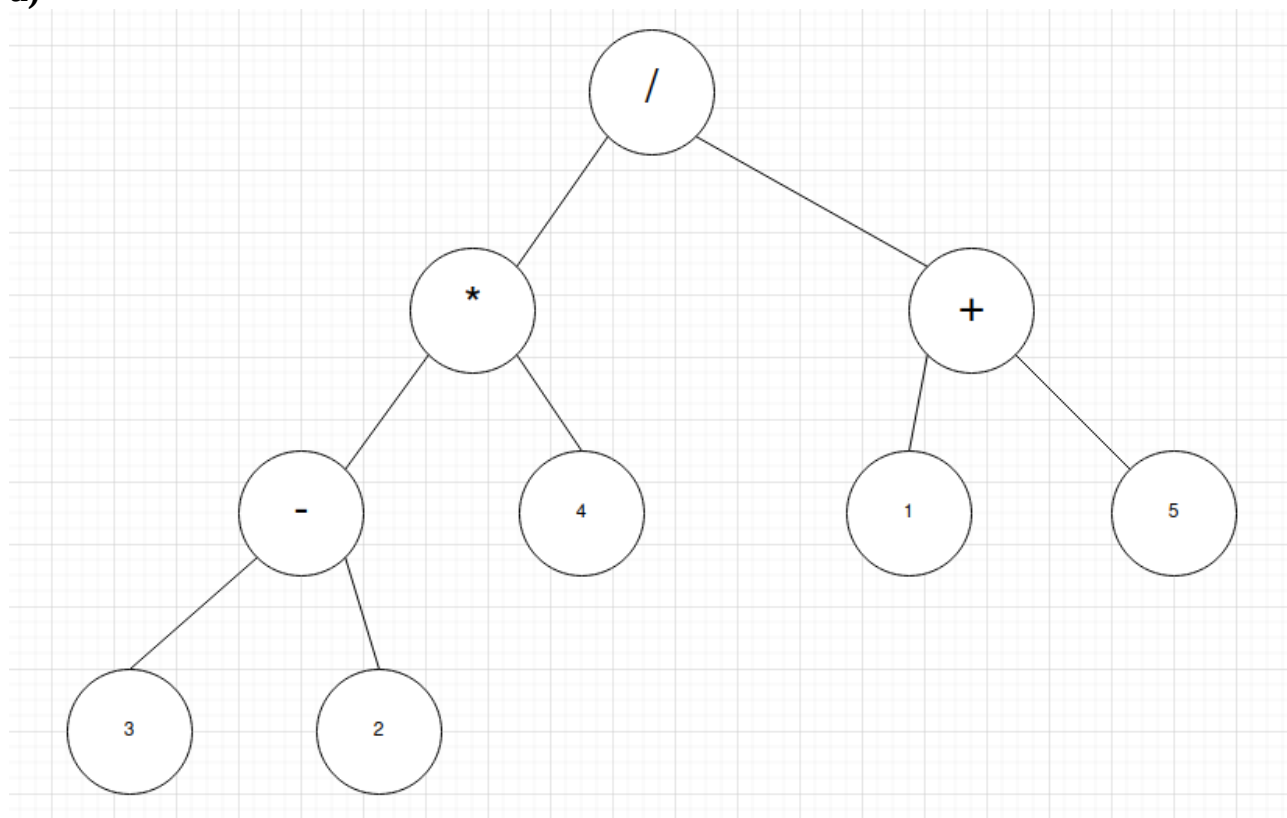
- Denne metoden kaller "settInnBakerst"  $n$  ganger. Siden "settInnBakerst" har en tidskompleksitet på  $O(n)$ , vil denne delen ha en tidskompleksitet på  $O(n^2)$ .
- Deretter kaller den "josephusProblem", som har en tidskompleksitet på  $O(n \times m)$ .
- Så totalt for denne metoden blir tidskompleksiteten  $O(n^2 + n \times m)$ . Imidlertid, for store verdier av  $m$ , vil  $n \times m$  dominere  $n^2$ , slik at den totale tidskompleksiteten tilnærmer seg  $O(n \times m)$ .

Samlet tidskompleksitet for utførelsen av utførJosephus-metoden (som representerer hoveddelen av programmet) er:  $O(n \times m)$

Dette betyr at tidskompleksiteten for programmet skalerer med både antall personer ( $n$ ) og lengden på intervallet ( $m$ ) mellom dem.

### Deloppgave 3

a)



b) Hvis jeg skal regne ut verdien av uttrykkstreet I a, ville jeg ha brukt postordentraversering.

c)  $((3-2)*4)/(1+5)$  Dette er postordentraversering.