

uke 14: Rekursjon

in1010 gruppe 1

Dagens plan

- Gjennomgang av ukens tema
- Quiz
- Oppgaver

Hva er rekursjon?

- Enkelt forklart: en metode som kaller på seg selv
- Mer definert: en teknikk som løser et problem ved å løse mindre problemer av samme type
- Kan ses på som en evig løkke

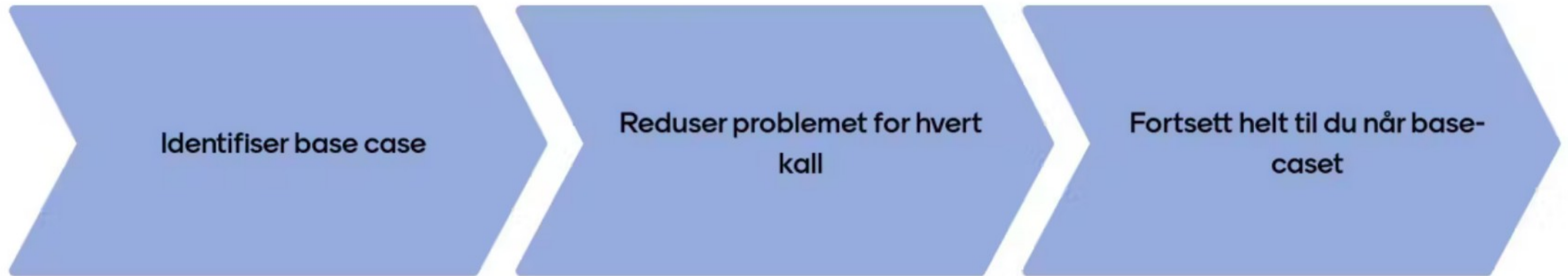
Hva må vi ha med i en rekursiv metode?

- Vi må ha et base tilfelle som stopper rekursjonen
- Noe som forandrer seg i kallene. - slik at vi til slutt når basistilfelle
- Rekursivt kall

```
void rekursjon(int x){  
    if(x == 0) return;  
    rekursjon(x-1);  
}
```

Base tilfelle

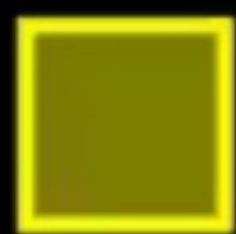
Rekursivt kall der noe forandrer seg i kallene



Prøv selv!

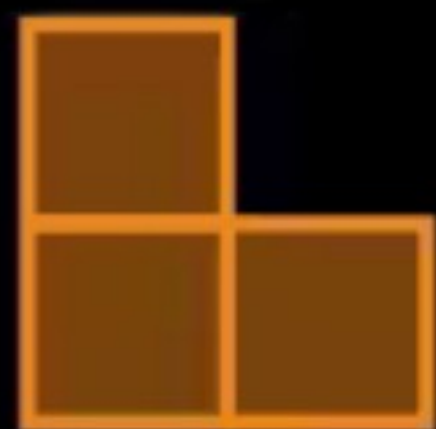
- Skriv en rekursiv metode som gitt en input n tar summen av alle positive heltall opptil n
- Tenk selv i 1 min. Skriv ned hvordan dere ville gjort det
- Diskuter med sidemann (ca 2-3 min)

$n = 1$



1

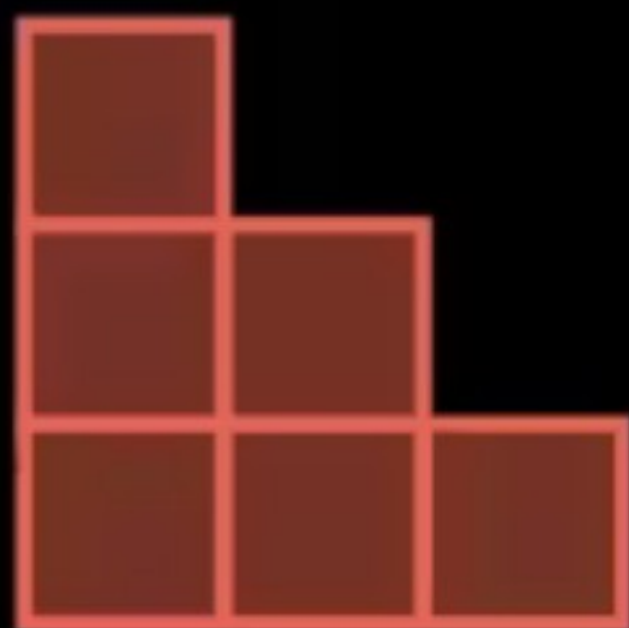
$n = 2$



$(1 + 2)$

3

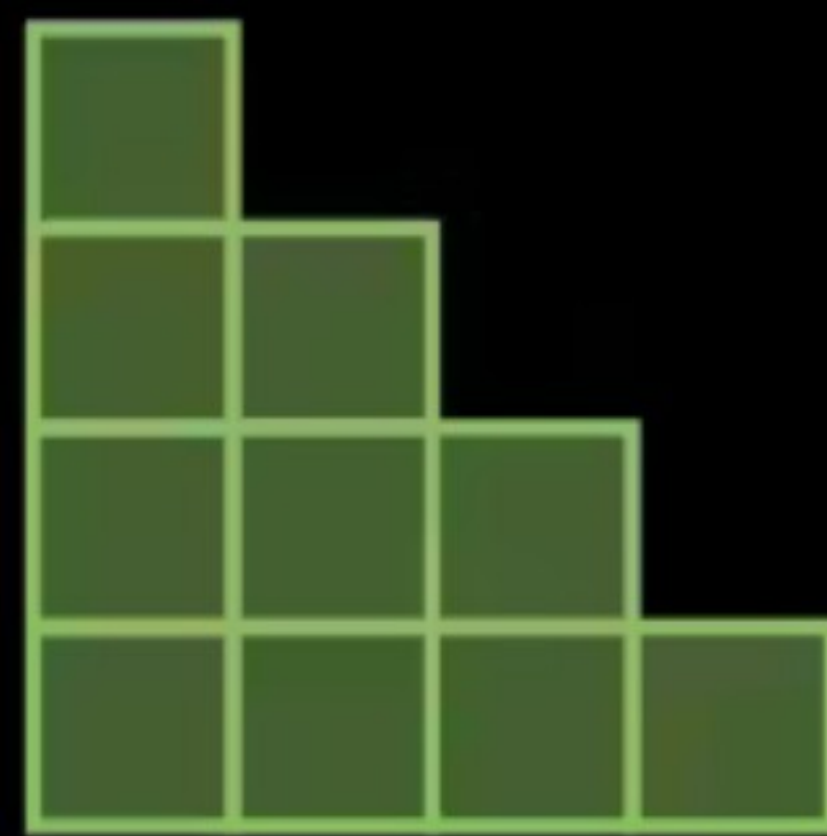
$n = 3$



$(1 + 2 + 3)$

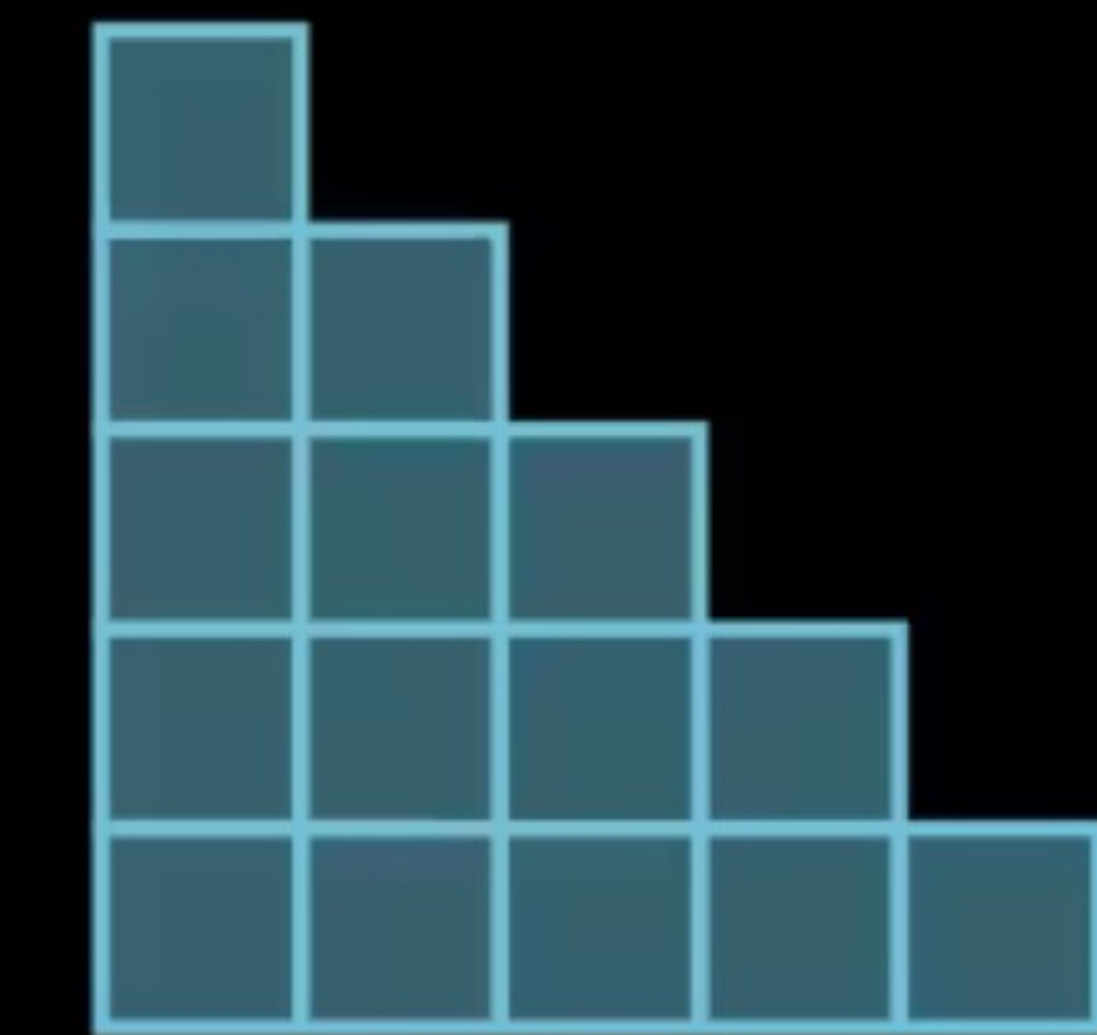
6

$n = 4$



$(1 + 2 + 3 + 4)$

10



$(1 + 2 + 3 + 4 + 5)$

```
int sum(int n){  
    if(n == 0) return 0;  
    return n + sum(n-1);  
}
```

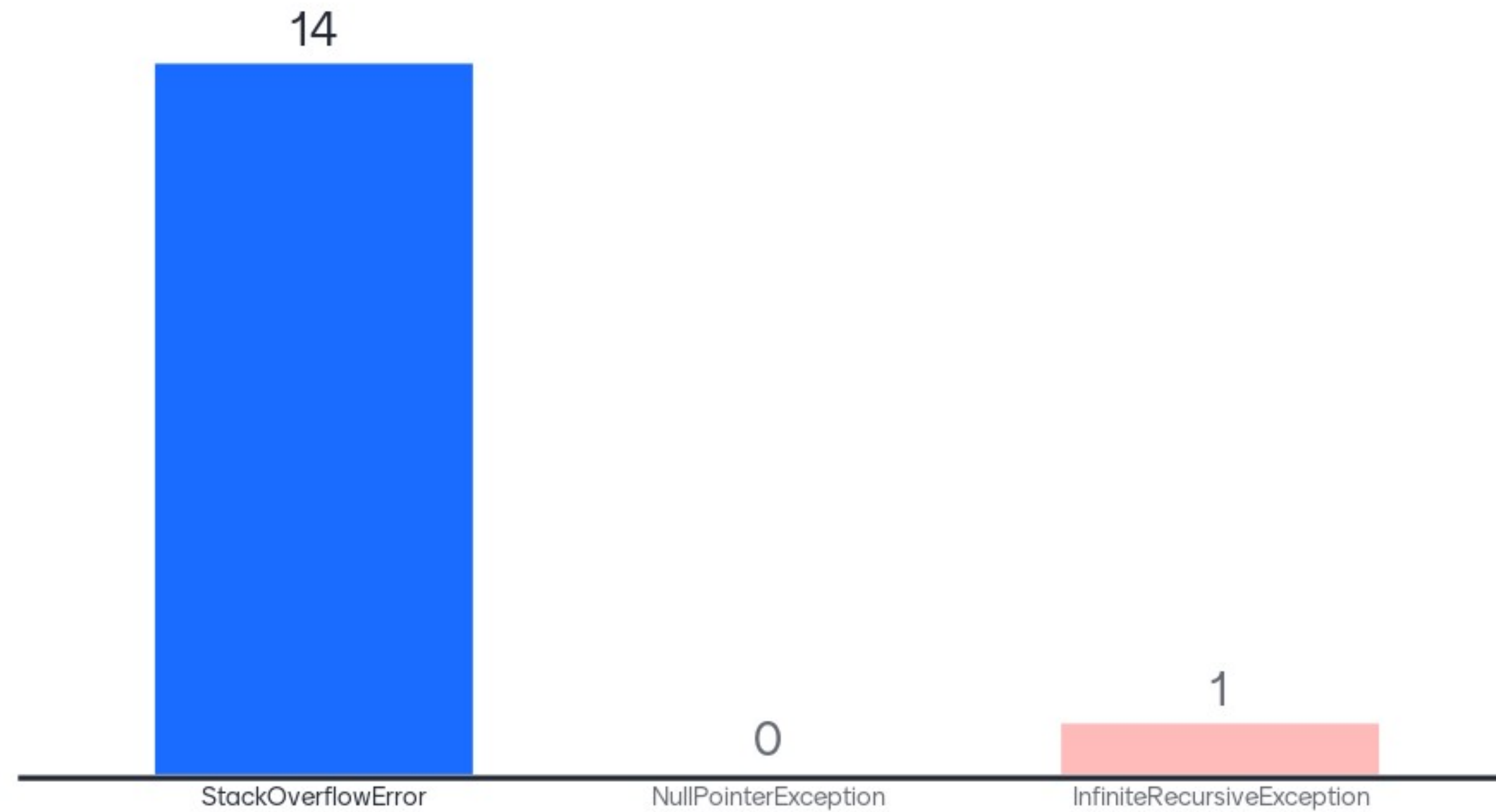
En mulig løsning, men hva skjer egentlig her?

Hva skjer?

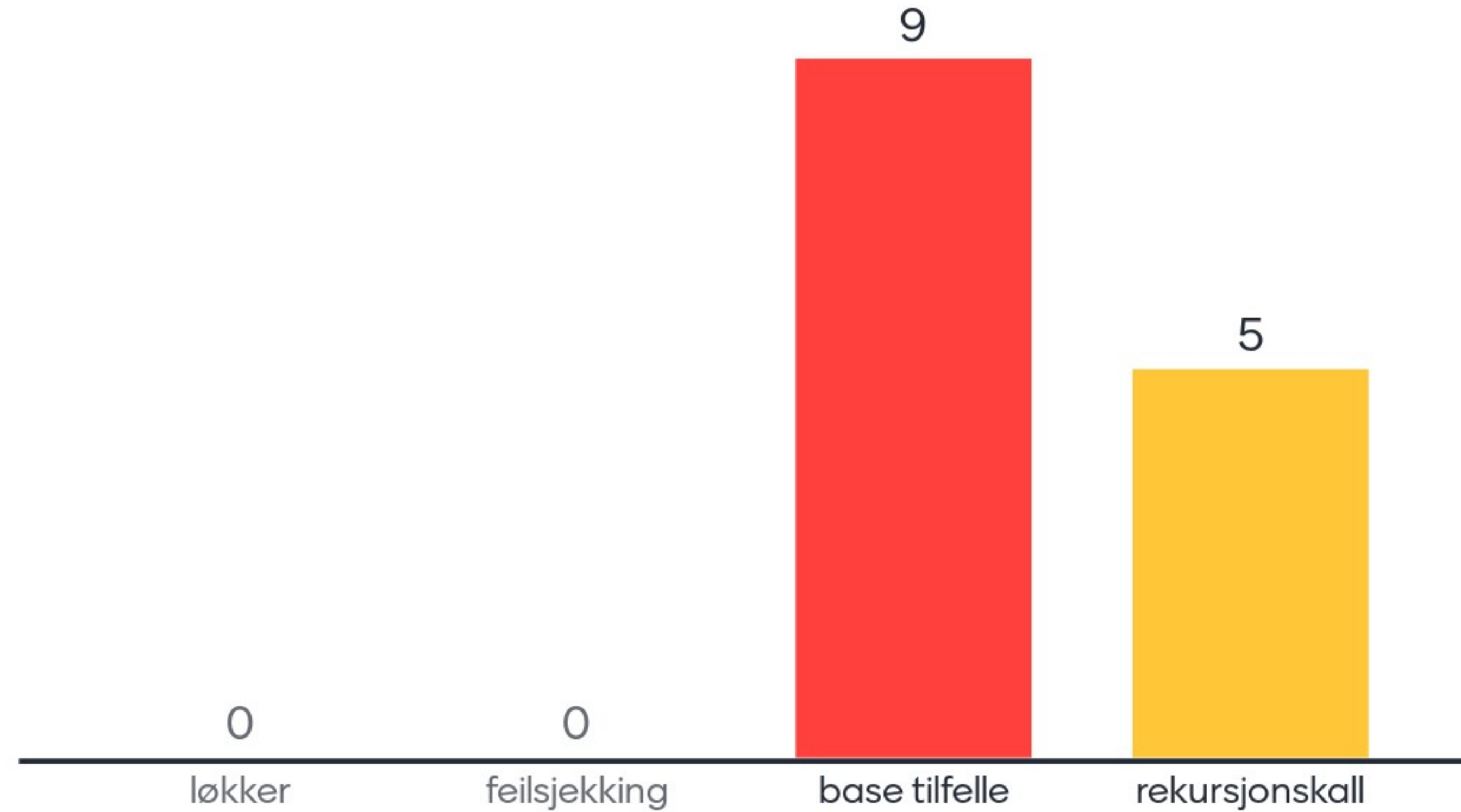
- La oss si vi kaller på sum(5)
- n er ikke lik 5 så vi returnerer 5+ sum(5-1)
- vi kaller på sum(4) men vi vet ikke hva det er, så da må vi beregne det. 4 er ikke 0 så sum(4) blir 4 + sum(3)
- må beregne sum(3) ... sum(0). 0 er lik 0 så 0 returneres
- returnerer riktig svar nedover i stack

```
int sum(int n){  
    if(n == 0) return 0;  
    return n + sum(n-1);  
}
```

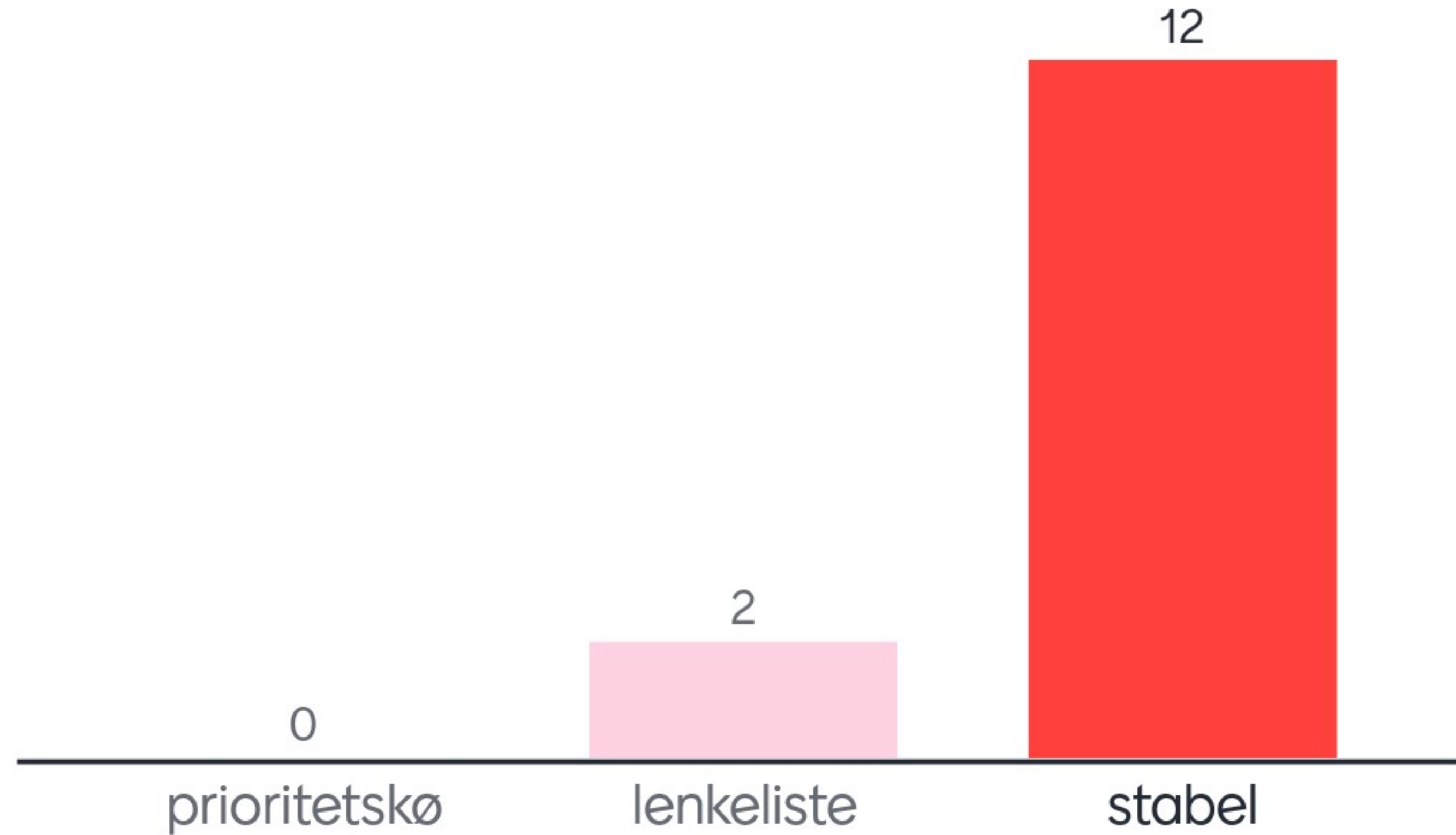

Uendelige rekursive kall (uten base tilfelle) kan generere



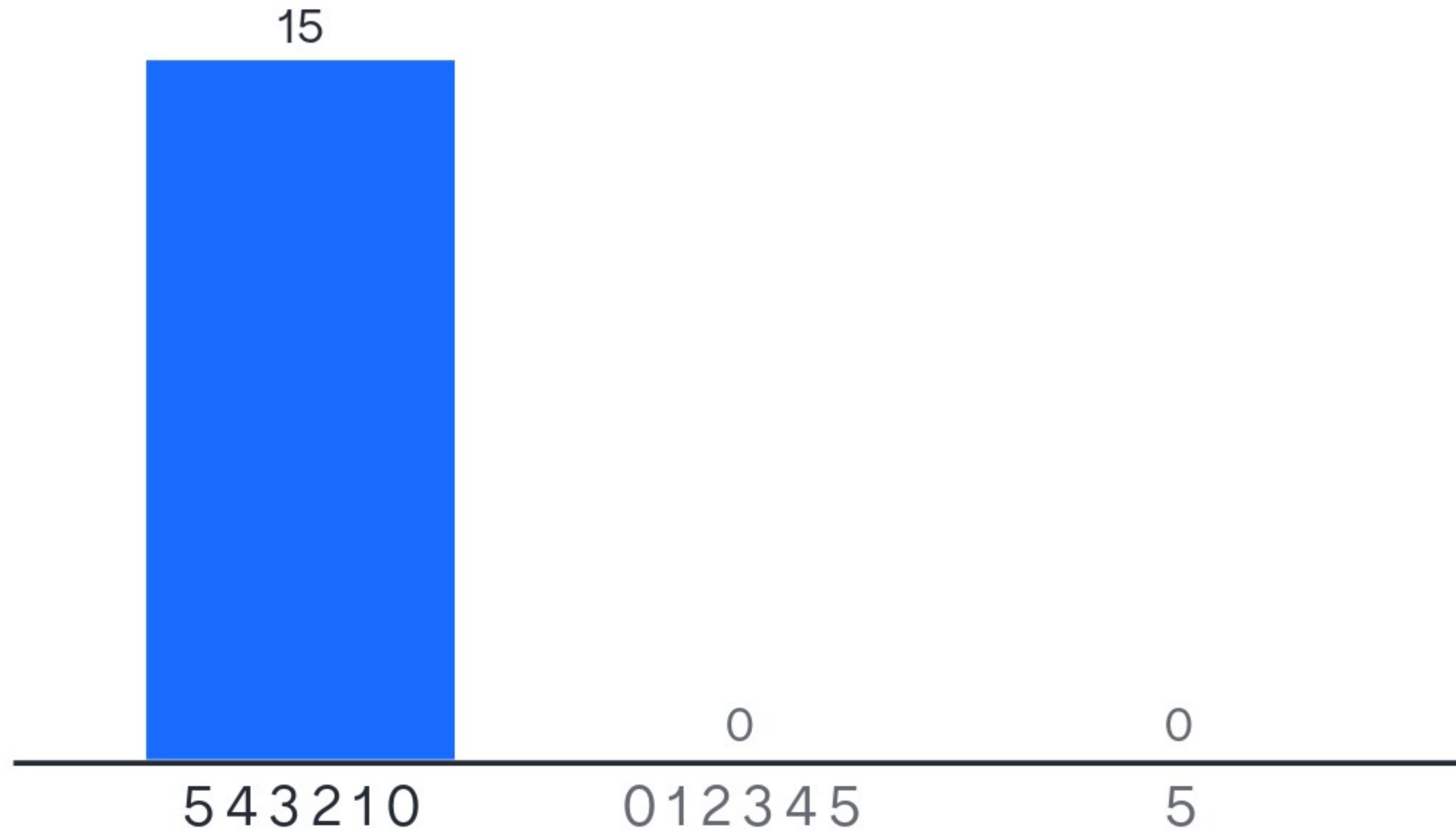
En rekursiv metode består av



Hvilken datastruktur er tett knyttet til rekursjon

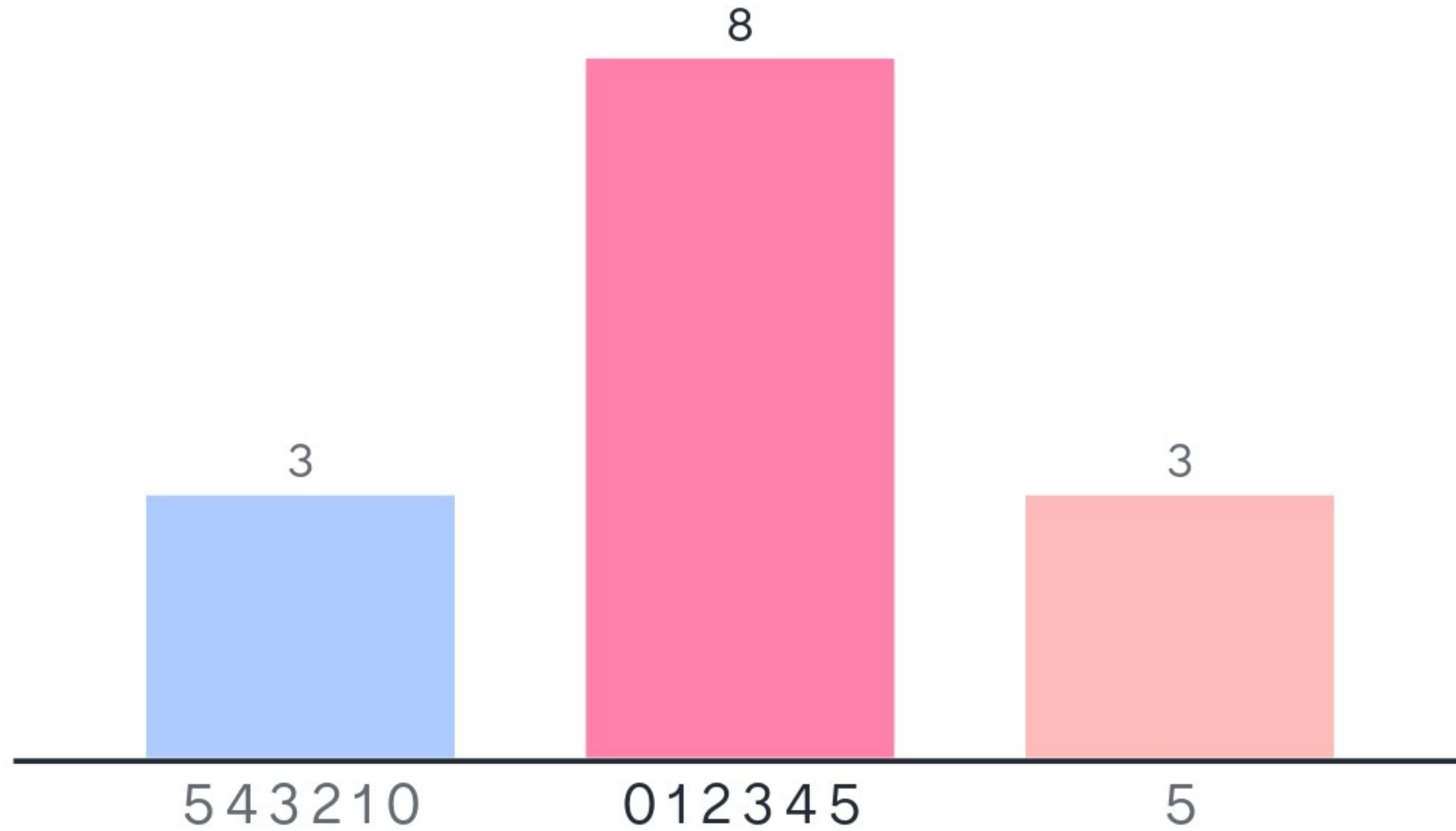


Hva printes ut?



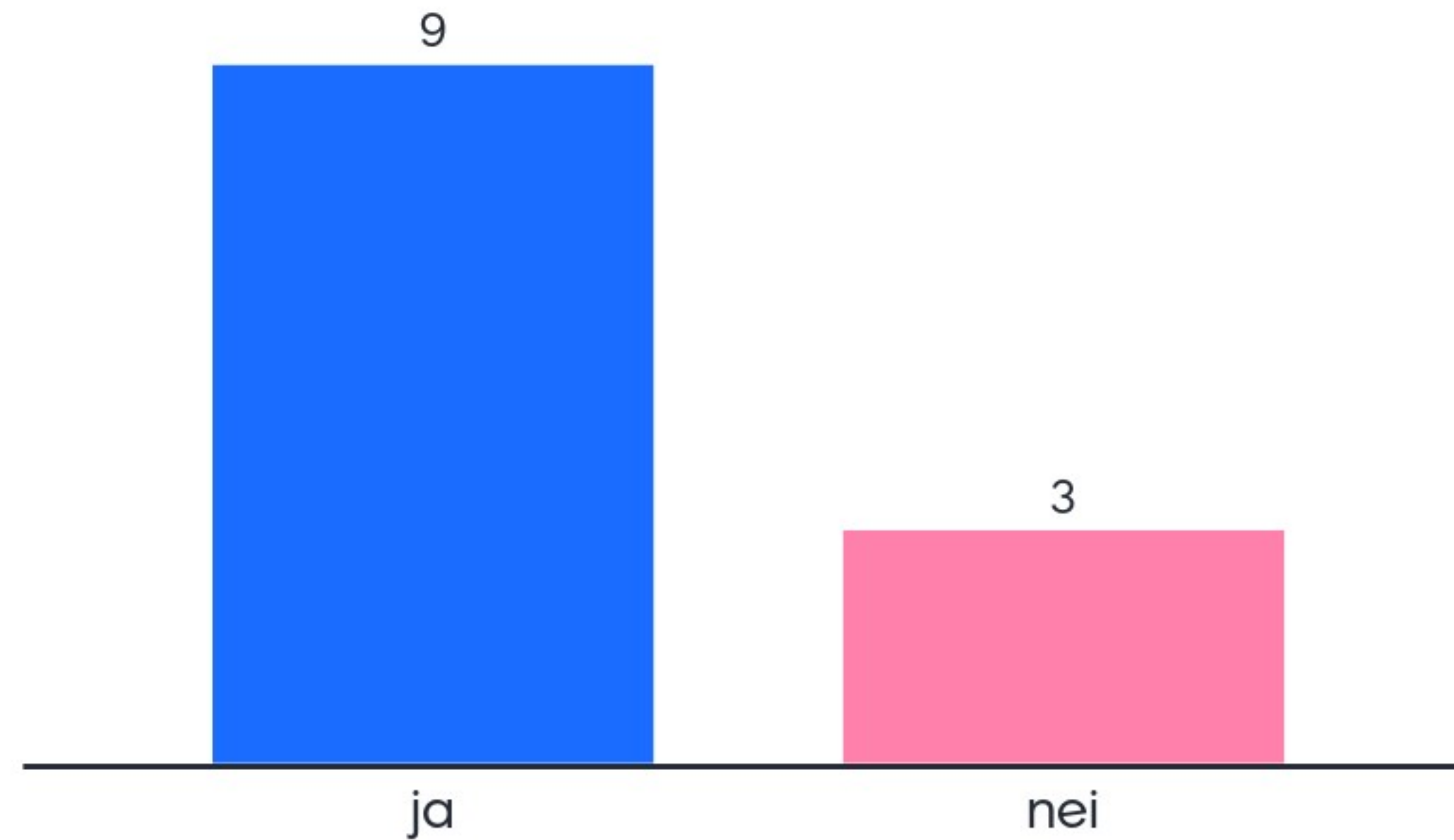
```
class Rekursjon{  
  
    static void skrivTall1(int n){  
        if(n < 0) return;  
        System.out.println(n);  
        skrivTall1(n-1);  
    }  
    Run | Debug  
    public static void main(String[] args) {  
        skrivTall1(n:5);  
    }  
}
```

Hva printes ut?



```
class Rekursjon{  
  
    static void skrivTall2(int n){  
        if(n < 0) return;  
        skrivTall2(n-1);  
        System.out.println(n);  
    }  
    Run | Debug  
    public static void main(String[] args) {  
        skrivTall2(n:5);  
    }  
}
```

Ønsker dere live koding der jeg viser et eksempel på hvordan man kan legge til elementer rekursivt i en lenkeliste?



Rekursjon er vanskelig

- Det hjelper å tegne
- Se på youtube
- Gjør oppgaver eller oblig