Uke12 – Rekursjon

in1010 - gruppe 2

22. april 2025

Hva er rekursjon?

- Når en funksjon kaller på seg selv for å løse problemmet.
- En rekursiv metode har
  - **et basistilfelle.** Når basistilfelle er nådd så stopper rekursjonen.
  - **rekursjonssteg.** Noe som forandrer seg slik at den stopper rekursjonen.

```
class Rekursjonstest {
    public static void main(String[] args) {
            tellNed(5);
    }
    public static void tellNed(int n){
            if (n < 0) return;
            System.out.println(n);
            tellNed(n-1);
    }
}
```

Basetilfellet

Rekursjonsstegt

Iterative løsninger

- Iterative løsninger og rekursjon gjør mye av de samme tingene, og iterative løsninger kan skriver om til rekursive løsninger og motsatt.
- Når skal man bruke hva?
  - **Iterative** løsninger blir brukt når operasjoner skal gjentas.
  - **Rekursjon** blir brukt for å løse problemmer.

```java
class Rekursjonstest {
    public static void main(String[] args) {
            tellNed(5);
    }
    public static void tellNed(int n){
            if (n < 0) return;
            System.out.println(n);
            tellNed(n-1);
    }
}
```

```java
class Iterasjonstest {
    public static void main(String[] args) {
            tellNed(5);
    }
    public static void tellNed(int n) {
            while(n >= 0) {
                System.out.println(n);
                n--;
            }
    }}
```

Oppgave: skriv den rekursiv metoden tellKaninOrer slik at outputen blir som forventet

KaninOreTeller.java > KaninOreTeller

```java
class KaninOreTeller {

    Run | Debug
    public static void main(String[] args) {
    System.out.println(tellKaninOrer(antallKaniner:0)); //Forventet resultat 0
    System.out.println(tellKaninOrer(antallKaniner:1)); //Forventet resultat 2
    System.out.println(tellKaninOrer(antallKaniner:2)); //Forventet resultat 4
    System.out.println(tellKaninOrer(antallKaniner:12)); //Forventet resultat 24
    System.out.println(tellKaninOrer(antallKaniner:234)); //Forventet resultat 486
    }
    public static int tellKaninOrer(int antallKaniner){

        //basetilfellet


        //rekursjonssteg

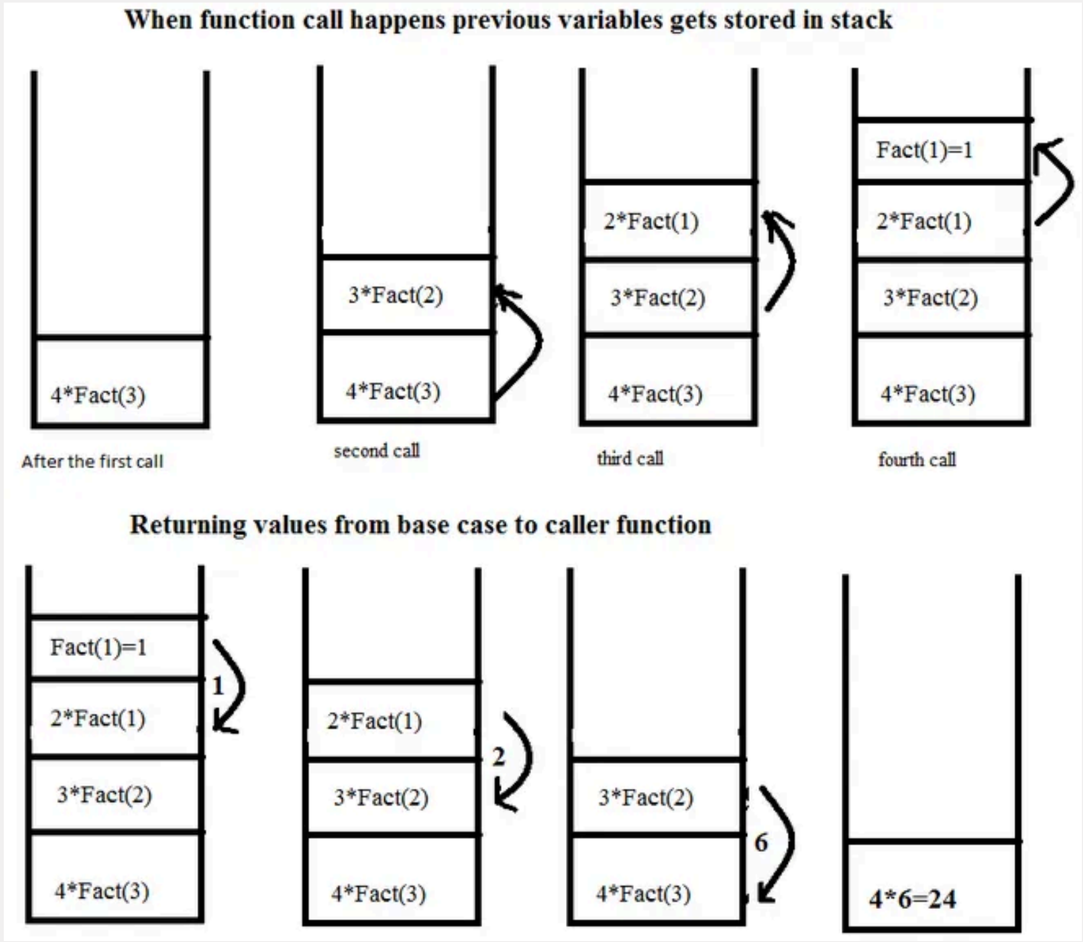
    }

}
```

Call stack i forbindelse med rekursjon

- Call stack er en **datastruktur** som blir brukt til å holde styr på funksjonskall.
- Når en funksjon blir kalt på, blir en boks(informasjon om argumenter, posisjon) lagt på stabelen.
- Call stacken operer i **LIFO** (Last In First Out), så det siste funksjonskallet er den første som blir ferdig.

Starter med Fact(4)



Seminaroppgaver