

## INF110 Ukeoppgaver: Uke 5

---

### OPPGAVE 1

---

Lag metoder for innsetting og fjerning av elementer i en toveis pekerkjede-liste.

### OPPGAVE 2

---

Fra MAW: 3.2, 3.17, 3.35

### OPPGAVE 3

---

Vi skal lage et program som behandler deklarasjonene i et Java-aktig program.

Vi vil representere deklarasjonene i programmet som lister. Programmet inneholder bare klasse-deklarasjoner og integer-deklarasjoner. Klassene kan igjen ha en liste av lokale deklarasjoner som også kan være klasser osv. Vi tenker oss at deklarasjonene i programmet er lest inn i en listestruktur der elementene ser slik ut:

```
class Dekl {
    String navn;
    Dekl neste;    // Neste deklarasjon i denne listen.
    Dekl lokale;  // For klasser: første lokale deklarasjon,
                // ellers null.
}
```

Første deklarasjon i programmet refereres av variabelen Dekl prog.

a) Skriv et Java-program som inneholder klasse- og integer-deklarasjoner der noen av klassene inneholder nye klasser osv.

b) Tegn den tilhørende listestrukturen.

c) Lag et program som skriver ut en deklarasjons-struktur som vi tenker oss eksisterer som en listestruktur i programmet. Det hele skal skrives som en sekvens av deklarasjonsnavn, der navnene på de lokale deklarasjonene i en klasse skal komme inne i en parentes, bak navnet på klassen.

Utskriften gjøres rekursivt ved at vi har en metode som skriver ut en liste, og som gjør rekursive kall for å skrive ut de lokale deklarasjonslistene.

#### OPPGAVE 4

---

Samme som oppgave 3, denne gangen løst med stakk.

Denne gangen skal vi bruke en stakk av Dekl-pekere for å holde orden på de deklarasjoner vi er i gang med å skrive ut, men enda ikke er ferdig med. Forsøk å få tak i likhetspunktene ved rekursive metoder og bruk av stakk.

Hvilke operasjoner trenger vi i den abstrakte datatypen stakk her? Skisser også to implementasjoner av denne stakken. Den ene skal bruke en array, og den andre skal bruke en listestruktur.

#### OPPGAVE 5

---

Vi skal skrive et program som sjekker visse aspekter ved Java-programmer, nemlig:

- at {}-par kommer riktig
- at vanlige parentes-par kommer riktig
- at det ikke kommer semikolon eller {} inne i parenteser

Vi skal ikke bry oss med detaljene i lesingen av programmet, men kan f.eks. anta at vi har en metode `String nesteSymb()` som leverer neste interessante symbol i programmet, og hvor "EOF" angir programslett.

Man kan bruke en (`String`) stakk for å sjekke dette. Beskriv et sett av stakk-operasjoner som er tilstrekkelig for denne sjekkingen, og skriv hovedprogrammet ut fra dette.

#### OPPGAVE 6

---

Vi skal lage et program som finner en måte å bevege en hest (springer) på sjakkbrett, slik at den er innom alle ruter en og bare en gang. En hest kan gå to skritt frem (i vilkårlig retning) og ett til siden. Oppgaven skal løses for et  $n \times n$  brett, og hesten skal starte i en gitt posisjon. HINT: Lag først en rekursiv prosedyre som genererer alle hesteturer av lengde  $n * n$  (også utenfor brettet). Legg så inn avskjæring slik at den:

- a) holder seg innenfor brettet
- b) aldri besøker en rute to ganger