

##  ving 2: Pekere og referanser

Else Lervik og Mildrid Ljosland

L restoffet er utviklet for faget IFUD1048 C++ for programmerere  
med grunnlag i l restoff fra LV195D Objektorientert programmering i C++

---

### Oppgaver

Oppgave 2 og 3 er teorioppgaver. Disse besvares i egen tekstfil, eventuelt pdf.

#### Oppgave 1

Du skal tegne, resonnere og programmere i denne oppgaven. Lever inn .cpp-programmet etter at oppgave b) er utf rt. Du trenger ikke levere inn tegninger.

a) F lgende er gitt:

```
int i = 3;
int j = 5;
int *p = &i;
int *q = &j;
```

Tegn opp variablene med innhold etter at disse definisjonene er utf rt.

Lag et lite program som skriver ut b de adressen til, og innholdet av disse fire variablene. Ved utskrift av adressen (eksempelvis, adressen til variabelen **j**: **cout << &j**) f r du skrevet ut adressen i 16-tallsystemet. Dersom du vil ha den ut i 10-tallsystemet, m  du caste: (**cout << (int) &j**).

Stemmer utskriften med tegningen din?

b) Vi g r ut fra at variablene har innholdet fra oppgave a. Hva blir skrevet ut n r f lgende programbit utf res?

```
*p = 7;
*q += 4;
*q = *p + 1;
p = q;
cout << *p << " " << *q << endl;
```

Tegn gjerne opp variablene etter hver enkelt setning. Utvid programmet fra a), og sjekk at det stemmer med tegningen din.

#### Oppgave 2

Teorioppgave.

Hva vil skje hvis du skriver

```
char *linje = nullptr; // eller char *linje = 0;
strcpy(linje, "Dette er en tekst");
```

### Oppgave 3

Teorioppgave.

Finn ting som kan gå galt i følgende programbit.

```
char tekst[5];
char *pekeren = tekst;
char letEtter = 'e';
cin >> tekst;
while (*pekeren != letEtter) {
    *pekeren = letEtter;
    pekeren++;
}
```

### Oppgave 4

Finn alle syntaksfeil i følgende programbit:

```
int a = 5;
int &b;
int *c;
c = &b;
*a = *b + *c;
&b = 2;
```

Legg setningene inn i en .cpp-fil og endre koden slik at den kompilerer. Beskriv årsaken til hver enkelt av kompileringsfeilene i kommentarer. Lever dette.

### Oppgave 5

Vi har definert variabelen **double tall**

Definer en peker som skal peke til **tall**.

Definer en referanse som skal referere til **tall**.

Vis tre måter å få tilordnet verdi til **tall** på.

Legg dette inn i en .cpp-fil som kompilerer.

### Oppgave 6

Du skal lage en funksjon og et hovedprogram som bruker denne funksjonen. Du skal bruke metoden som er beskrevet i eksempel 1 i kapittel 1.7.1.

Lag en funksjon som finner summen av elementene i en tabell. Funksjonen skal ha følgende hode:

```
int finnSum(const int *tabell, int antElementer)
```

I hovedprogrammet skal du opprette en tabell med 20 elementer, fyll tabellen med tallene 1, 2, 3, osv. og bruk funksjonen til å finne summen av

- a) De 10 første tallene
- b) De 5 neste tallene
- c) De 5 siste tallene