

C++ vs Java: (for vår del)

- Funksjoner kan også ligge utenfor klasser, bl.a. `main`
- `cout <<` - `System.out.println(.....)`
`cin >>` - `System.in.....`
- Ikke destructor
- `struct` (= klasse med *kun* public datamedlemmer)
- `private:`
`public:` *en gang* vs. *pr.datamedlem/funksjon*
- `const` (som parameter) = `final`
- `const`-funksjoner i klasser – kan *ikke* endre datamedlemmer
- *Ikke* Templates (men har noe lignende: Generics)
- *Ikke* referanseoverføring (`&`)
- Pekere (`new`, `delete`, `.` vs `->`)
I C++ må vi skille på om har objekt (`.`) eller *peker* til et (`->`) :

- class A { }

Java:

```
A aobj; // Referanse-variabel
aobj = new A; // Objekt lages
```

Bruk:

```
aobj.xxxxx
```

C++:

```
A aobj; // Objekt lages!
A* aobj2; // Peker -variabel
aobj2 = new A; // Objekt lages
```

```
aobj.xxxxx
aobj2->xxxxx
```

- (mer vesentlig?)

Noen få biblioteksfunksjoner emnets kodeeksempler bruker:

- <cstring>:
 - strlen(...) - en char-arrays tekstlengde
 - strncmp(..., ..., x) - sammenligner 'x' tegn i to tekster
 - strstr(..., ...) - leter etter en subtekst i en annen
- <cstdlib>:
 - itoa(..., ..., 10) - konverterer tall til tekst
 - rand() / srand(0) - sørger for tilfeldig talltrekking
- <iomanip>:
 - setw(...) - neste utskrift gjøres høyrejustert
- <cctype>:
 - toupper(...) - gjør om en bokstav (a-z) til stor
 - isalpha(...) - er et tegn en bokstav (A-Z, a-z)?
- <cmath>:
 - sqrt(...) - finner kvadratroten av et tall