

Lektion 1 – Grunderna

1 C++ 101

Ett program har alltid följande struktur:

```
#include <bits/stdc++.h>
using namespace std;

int main() {

    return 0;
}
```

Man skriver sedan själva programmet innanför måsparenteserna inuti *main-funktionen*. Man kan också skriva utanför, men det kommer vi in på i senare lektioner. Alla rader avslutas med ; (ett semikolon).

2 Output

Utskrift görs genom

```
cout << "Hej" << endl;
```

Detta kommer skriva ut Hej. Man kan också ta bort `endl`, och då kommer den inte hoppa till en ny rad.

3 Variabler

Man kan skapa variabler av olika typer. En variabel kan man se som en container man sparar saker i, liknande variabler från matematiken. Man skapar en variabel genom att först skriva vilken typ av variabel det är, och sen skriva vilket namn man vill ge variabeln

3.1 Variabeltypen `int`

```
int x;
```

skapar en variabel av typen `int`, som heter `x`. Att det är en `int` betyder att det är ett heltal. För att tilldela `x` ett värde skriver man t.ex.

```
x = 5;
```

som kommer göra så att `x` sparar värdet 5. För att skriva ut vad variabeln sparar kan man sedan använda

```
cout << x << endl;
```

som i detta fall kommer skriva ut 5.

Man kan använda de vanliga matematiska operationerna addition (+), subtraktion (−), multiplikation (·) och division (÷). Detta skrivs i C++ som +, −, * och /. Exempelvis kommer

```
cout << 5 + 3 << endl;  
cout << 5 * 3 << endl;  
cout << 5 - 3 << endl;  
cout << 8 / 4 << endl;
```

skriva ut 8, 15, 2, och 2. Operatorerna fungerar också på variabler på samma sätt. Man skulle alltså i vårt fall kunna byta ut femmorna mot `x`. Man kan också ändra en variabel. Om vi har `x` som var lika med 5, så kan vi skriva

```
int x = 5;  
x = 12;
```

Det är alltid den senaste raden som räknas, och `x` kommer alltså innehålla talet 12. Till exempel kommer följande

```
int x = 5;  
cout << x << endl;  
x = 12;  
cout << x << endl;  
x = x + 5;  
cout << x << endl;
```

att skriva ut 5, 12 och 17. Näst sista raden betyder att vi ökar `x` med 5.

3.2 Variabeltypen `string`

Det finns ytterligare en variabeltyp som heter `string`, som sparar text. Exempelvis kan man skriva

```
string namn;
```

Man tilldelar `namn` ett värde genom

```
namn = "Teodor";
```

Notera att man måste ha citationstecken runt text för att göra det till en `string`.

4 Input

För att mata in ett värde i en variabel skriver man

```
int x;  
cin >> x;
```

skriver man nu en femma i inputfönstret kommer `x` få värdet 5 när man kör programmet. Samma sak kan göras med en `string`.

5 Att lösa ett problem

På <https://turinggang.se> kan du lösa problem. Du kan skriva ett program i editorn (<https://turinggang.se/editor>), och när du tror att du fått rätt på det så kan du skicka in din lösning för automatisk rättning. Det som då händer är att hemsidan tar din kod, kör den på hemliga testfall, och avgör om ditt program ger rätt svar.

Du skickar in din lösning såhär: gå till problemet och tryck på den gröna Submit-knappen. I textrutan som då kommer upp ska du klistra in all text som ditt program består av i editorn (d.v.s. från och med `#include <bits/stdc++.h>` till `return 0; }`). När du sedan trycker på Submit på den här sidan så skickar du in ditt program. Efter ett litet tag kommer du se resultatet när du laddar om sidan du hamnat på.

Du kommer få någon av följande bedömningar på ditt program:

Running...	Ditt program körs fortfarande. Ladda om sidan för att se resultatet.
Compile Error	Ditt program innehåller ett fel som gör att det inte kan köras. Kopierade du allting från editorn?
Run Time Error	Ditt program kraschade när det körde. Testa i editorn med stora tal eller konstiga fall.
Time Limit Exceeded	Ditt program är för långsamt (men kanske har rätt svar).
Wrong Answer	Ditt program kan köras men skriver ut fel svar. Kan bero på att du tänkt fel.
Accepted	Du klarade alla testfall. Grattis!

Notera att för att få en lösning Accepted så måste ditt program få rätt svar och vara inom tidsgränsen på **alla testfall**.

6 Exempel

Här är ett exempelprogram som läser in ett tal **x**, lägger till 5 till det, och skriver ut svaret.

```
#include <bits/stdc++.h>
using namespace std;

int main() {

    int x;
    cin >> x;
    x = x + 5;
    cout << x << endl;

    return 0;
}
```