

Klasser och Objektorientering skiljer c++ från c

Rekapitulera funktioner

Referensanrop, överlagring, förvalda värden

Funktioner

Om vi definierar funktioner efter main()

måste de deklaras före main()

Så har vi gjort tidigare

Man kan också definiera funktionen före main() då krävs ingen separat definition.

Referensanrop:

```
float x=2;
```

```
float y=4;
```

```
float z=3;
```

```
float max(float tal1, float tal2);
```

```
max(x,y); //Värdeanrop (värdena av x och y kopieras till tal1 och tal2);
```

```
void matain( float & tal1, float & tal2); // referensanrop refererar till minnesplatser för x y;
```

```
matin(x,y);
```

Funktionsöverlagring eller Funktionspolymorfism.

i C++ kan funktioner ha samma namn om antalet parametrar skiljer:

```
float max(float tal1, float tal2);
```

```
float max(float tal1, float tal2, float tal3);
```

```
max(y,z);
```

```
max(x,y,z);
```

systemet förstår av antalet parametrar vilken funktion som anropas;

Förvalda värden i funktioner

c++ tillåter att förvalda värden anges i parameterlistan

```
void matain(float & tal1, float & tal2, char * prompt=" ? ");
```

```
// char * gammaldags sträng
```

```
//void matain(float & tal1, float & tal2, string prompt=" ? "); // = samma sak
```

Det tredje argumentet är förvalt om anropet inte använder den tredje parameter kommer det förvalda värdet att användas.

Se exemplet nedan:

```

#include <cstdlib>
#include <iostream>

using namespace std;
void matain(float & tall1,float & tal2, char * prompt=" ? ");
//void matain(float & tall1,float & tal2, string prompt=" ? ");
//string == char *

int main(int argc, char *argv[])
{
    float x,y;

    matain(x,y);

    cout<<"Vardet for x = "<<x<<" vardet for y = "<<y<<endl;

    matain(x,y,"ge varde for x och y");

    cout<<"Vardet for x = "<<x<<" vardet for y = "<<y<<endl;

    system("PAUSE");
    return EXIT_SUCCESS;
}

void matain(float & tall1,float & tal2, char * prompt)
//void matain(float & tall1,float & tal2, string prompt)
{
    cout<<prompt;
    cin>>tall1>>tal2;
}

```

Kör gärna exemplet ovan för att se vad som händer!

UPPGIFT 1

```

#include <cstdlib>
#include <iostream>
#include <conio.h>

using namespace std;

int main(int argc, char *argv[])
{
    int timme,minut,sekund;
    cout<<"Mata in timme, minut och sekund";
    cin>>timme>>minut>>sekund;
    cout<<timme<<": "<<minut<<": "<<sekund;
    getch();
    return EXIT_SUCCESS;
}

```

Skriv om programmet ovan med hjälp av två void_funktioner matain() och visa()

Använd biblioteket iomanip, och manipulatorerna setfill() och setw() för att få utskriften så här: **02:44:06**

UPPGIFT 2

Förbättra koden , om du inte redan gjort det, så att användaren endast kan skriva in rimliga värden för timme, minut och sekund, d.v.s. värde för timme kan inte vara 27!

UPPGIFT 3

Gör nu en funktion som beräknar hur många sekunder som förflutit sedan dygnets början. Anropa funktionen efter utskriften av timmar, minuter och sekunder.

UPPGIFT 4

Gör nu en funktion "slumpaKlockSlag" som använder biblioteket `ctime` samt funktionerna `srand` och `rand`

UPPGIFT 5 (överkurs)

Skriv en funktion som returnerar aktuellt datum och aktuell tid. Du kan använda den inbyggda strukturen `tm`. Här krävs nog en del sökande och tolkande av information på Internet, men börja med att gogla med sökorden `struct tm` och `c++` och se om du hittar tillräcklig info för att lösa uppgiften (jobba gärna två och två).