programming in C++

Jonas Vejlin

Who I am

- Background: M.Sc in software engineering at AAU in 2009
- IT developer: Working with modeling at Foulum 2009-20??
- Working with FASSET, Animal Change, Nitroscape, C-tool
- Java, C++, C#
- VBA, R, Matlab

Goal

Goal

- Automate some task

Goal

- Automate some task
- Implement some statistic method to analyst data

Goal

- Automate some task
- Implement some statistic method to analyst data
- Extract the necessary information from file

Goal

- Automate some task
- Implement some statistic method to analyst data
- Extract the necessary information from file
- Modifier larger models such as Daisy or Fasset

Parts

Part 1 (Today)

Basic programming

Part 2

Control structure such as loops and if-else

Part 3

Vector, Files and Functions

C++

C++

- General purpose programming language

C++

- General purpose programming language
- Both high-level and low-level language features

C++

- General purpose programming language
- Both high-level and low-level language features
- Provide possibility for Object-oriented programming

C++

- General purpose programming language
- Both high-level and low-level language features
- Provide possibility for Object-oriented programming
- Needs to compiled

C++

- General purpose programming language
- Both high-level and low-level language features
- Provide possibility for Object-oriented programming
- Needs to compiled
- Created by Bjarne Stroustrup at Aarhus Uni

Programming

Programming

- Define the problem

Programming

- Define the problem
- Write algorithm that solves the problem

Programming

- Define the problem
- Write algorithm that solves the problem
- Program the algorithm

Programming

- Define the problem
- Write algorithm that solves the problem
- Program the algorithm
- Test the program

Programming

- Define the problem
- Write algorithm that solves the problem
- Program the algorithm
- Test the program
- Make the computer do all the hard work

Termologi

Termologi

- Programs have to be translated to the target computers machine language

Termologi

- Programs have to be translated to the target computers machine language
  - Compiler: the program that translates

Termologi

- Programs have to be translated to the target computers machine language
  - Compiler: the program that translates
  - Source file: input to the compiler

Termologi

- Programs have to be translated to the target computers machine language
  - Compiler: the program that translates
  - Source file: input to the compiler
  - If the program is syntactically correct, the compiler will save the machine language instructions in an object file

Termologi

- Programs have to be translated to the target computers machine language
  - Compiler: the program that translates
  - Source file: input to the compiler
  - If the program is syntactically correct, the compiler will save the machine language instructions in an object file
- The linker combines an object file with already existing libraries of functions and procedures in an executable file

Hallo World

Source Code

```
// my first program in C++
 #include <iostream>
 using namespace std;
 int main()
{
     cout << "Hallo World!";
     cin.get();
 }
```

Output

Hallo World

Source Code

```
// my first program in C++
 #include <iostream>
 using namespace std;
 int main()
{
     cout << "Hallo World!";
     cin.get();
 }
```

Output

Hallo World

Source Code

```
// my first program in C++
 #include <iostream>
 using namespace std;
 int main()
{
     cout << "Hallo World!";
     cin.get();
 }
```

Output

Hallo World

Source Code

```
// my first program in C++
 #include <iostream>
 using namespace std;
 int main()
{
     cout << "Hallo World!";
     cin.get();
 }
```

Output

Hallo World

Source Code

```
// my first program in C++
 #include <iostream>
 using namespace std;
 int main()
{
     cout << "Hallo World!";
     cin.get();
 }
```

Output

Hallo World

Source Code

```
// my first program in C++
 #include <iostream>
 using namespace std;
 int main()
{
      cout << "Hallo World!";
      cin.get();
 }
```

Output

Hallo world!

Hallo World

Source Code

```
// my first program in C++
 #include <iostream>
 using namespace std;
 int main()
{
     cout << "Hallo World!";
     cin.get();
 }
```

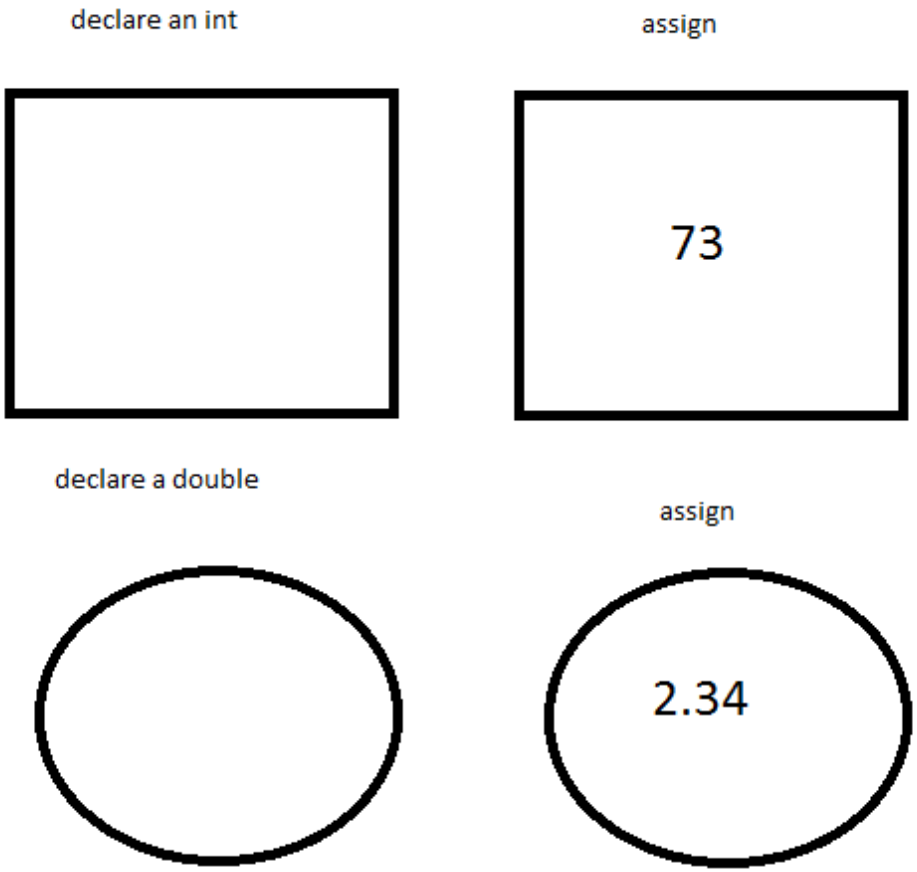
Output

Hallo world!

Basic Types

Group	Type names
Integer types	(signed) **int**
	unsigned int
	long
Floating-point	(signed) **double**
	unsigned double
	long double
Character types	char

Declare And Assign



Declare And Assign

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double decimalPoint;
    decimalPoint = 0.7;
    int interger=0;
    interger=2;
    cout <<"the value of
decimalPoint"<<endl;
    cout <<
decimalPoint<<endl;
    cout << interger<<endl;
cin.get();
}
```

Output

data

decimalPoint

Declare And Assign

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double decimalPoint;
    decimalPoint = 0.7;
    int interger=0;
    interger=2;
    cout <<"the value of
decimalPoint"<<endl;
    cout <<
decimalPoint<<endl;
    cout << interger<<endl;
cin.get();
}
```

Output

data

decimalPoint=0.7

Declare And Assign

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double decimalPoint;
    decimalPoint = 0.7;
    int interger=0;
    interger=2;
    cout <<"the value of
decimalPoint"<<endl;
    cout <<
decimalPoint<<endl;
    cout << interger<<endl;
cin.get();
}
```

Output

data

decimalPoint=0.7

interger=0

Declare And Assign

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double decimalPoint;
    decimalPoint = 0.7;
    int interger=0;
    interger=2;
    cout <<"the value of
decimalPoint"<<endl;
    cout <<
decimalPoint<<endl;
    cout << interger<<endl;
cin.get();
}
```

Output

data

decimalPoint=0.7

interger=2

Declare And Assign

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double decimalPoint;
    decimalPoint = 0.7;
    int interger=0;
    interger=2;
    cout <<"the value of
decimalPoint"<<endl;
    cout <<
decimalPoint<<endl;
    cout << interger<<endl;
cin.get();
}
```

Output

```
the value of
decimalPoint
```

data

decimalPoint=0.7

interger=2

Declare And Assign

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double decimalPoint;
    decimalPoint = 0.7;
    int interger=0;
    interger=2;
    cout <<"the value of
decimalPoint"<<endl;
    cout <<
decimalPoint<<endl;
    cout << interger<<endl;
cin.get();
}
```

Output

```
the value of
decimalPoint
0.7
```

data

decimalPoint=0.7
interger=2

Declare And Assign

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double decimalPoint;
    decimalPoint = 0.7;
    int interger=0;
    interger=2;
    cout <<"the value of
decimalPoint"<<endl;
    cout <<
decimalPoint<<endl;
    cout << interger<<endl;
cin.get();
}
```

Output

```
the value of
decimalPoint
0.7
2
```

data

decimalPoint=0.7
interger=2

Statements

3 different kinds

- Expression statement
- Compound statement
- Control statement

Expression statements

- An expression statement consists of an expression followed by a semi colon
- The execution of such an expression implies the evaluation of the related expression
- Eg:
  - a = 6;
  - c = a + b;
  - ; (empty statement)

Compound statements

- Consists of several individual statements enclosed by { }
- Whatever lies inside { } is to be interpreted as a single statement
- Also called scope
- variables declared inside a Scope cannot be seen from the outside
- Eg:

```
{
        statements1;
        statements2;
}
```

Control statements

- These control the flow of execution in a program or a function
- 2 kinds

Control statements

- These control the flow of execution in a program or a function
- 2 kinds
  - Selection
    - if, if-else, switch

Control statements

- These control the flow of execution in a program or a function
- 2 kinds
  - Selection
    - if, if-else, switch
  - Repetition
    - for, while, do-while

Operator Precedence

Operators	How to write them
multiplicative	* / %
additive types	+ -
relational	< > <= >=
equality	== !=
logical AND	&&
logical OR	||
assignment	= += -= *= /= %=

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    cout<<"Adding";
    result=i + j;
    cout<<" i + j = ";
    cout<<result<<endl;
cin.get();
}
```

Output

data

i=10

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    cout<<"Adding";
    result=i + j;
    cout<<" i + j = ";
    cout<<result<<endl;
cin.get();
}
```

Output

data

i=10

j=20

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    cout<<"Adding";
    result=i + j;
    cout<<" i + j = ";
    cout<<result<<endl;
cin.get();
}
```

Output

data

i=10

j=20

result

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    cout<<"Adding";
    result=i + j;
    cout<<" i + j = ";
    cout<<result<<endl;
cin.get();
}
```

Output

Adding

data

i=10

j=20

result

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    cout<<"Adding";
    result=i + j;
    cout<<" i + j = ";
    cout<<result<<endl;
cin.get();
}
```

Output

Adding

data

i=10

j=20

result

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    cout<<"Adding";
    result=i + j;
    cout<<" i + j = ";
    cout<<result<<endl;
cin.get();
}
```

Output

Adding

data

i=10

j=20

result=30

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    cout<<"Adding";
    result=i + j;
    cout<<" i + j = ";
    cout<<result<<endl;
cin.get();
}
```

Output

Adding i + j =

data

i=10

j=20

result=30

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    cout<<"Adding";
    result=i + j;
    cout<<" i + j = ";
    cout<<result<<endl;
cin.get();
}
```

Output

Adding i + j = 30

data

i=10

j=20

result=30

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

data

i=10

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

data

i=10

j=20

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

data

i=10

j=20

result

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

data

i=10

j=20

result

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

data

i=10

j=20

result

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

data

i=10

j=20

result=30

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

i / j+1 = 1.5

data

i=10

j=20

result=30

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

i / j+1 = 1.5

data

i=10

j=20

result=30

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

i / j+1 = 1.5

data

i=10

j=20

result=30

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

i / j+1 = 1.5

data

i=10

j=20

result=30

Operators

Source Code

```
#include <iostream>
using namespace std;
int main()
{
    double i = 10;
    double j = 20;
    double result;
    result=i / j+1;
    cout<<" i / j+1 = " <<
result<<endl;
    result=i / (j+1);
    cout<<" i / (j+1) = "<<
result<<endl;
cin.get();
}
```

Output

i / j+1 = 1.5
i / (j+1) = 0.47619

data

i=10
j=20
result=30

Problems

Source Code

```
#include <iostream>
int main()
{
    int a=5;      int a=0;
    int result=a/b;
    co ut<<result<<endl
    cout<<"hel lo
world"<<endl;
    int sum=2-2;
    cout<<"the sum of 2+2 is:
";
    cout<<sum<<endl;
```