

PROGRAMMING LANGUAGES 1

Introduction



HELLO WORLD!

```
#include <iostream>
using namespace std;

// This is a comment
int main()
{
    cout << "Hello World!";
    return 0;
}
```



HELLO WORLD!

```
#include <iostream>
using namespace std;
```

← Preprocessor directive

← Use as-is for now

```
// This is a comment
```

← Comment

```
int main()
```

← Main function header

```
{
```

← Opening brace

```
    cout << "Hello World!";
```

← Code line to print output

```
    return 0;
```

} Function body

```
}
```

← Closing brace



PROGRAM EXECUTION



INTEGRATED DEVELOPMENT ENVIRONMENT (IDE)

- An application that includes all tools needed for programming
 - Editor
 - Compiler
 - Builder
 - Executer
 - Debugger
- We will be mainly using **Eclipse** for the course
 - Feel free to use any other IDE



PREPROCESSOR DIRECTIVES

- Outside the body of the function
- Tells the compiler to perform task before processing the program
- Not a real part of the program to be executed
- Examples:
 - `#include <iostream>` → Include the standard IO-stream library
 - `#define PI 3.14` → Define a constant PI



#include

#include <iostream>

#include <iostream.h>

#include "myfile.h"

→ Include a standard library

→ Include a built-in header file

→ Include a file in the same directory



#define

```
#include <iostream>
```

```
#define PI 3.14
```

```
int main()
```

```
{
```

```
    double r = 10;
```

```
    double area = PI * r * r; // Compiler replaces PI by 3.14
```

```
    cout << area << endl;
```

```
    r = 5;
```

```
    area = PI * r * r;
```

```
    cout << area << endl;
```

```
    return 0;
```

```
}
```



LINES

- Each line must end with a semicolon ;
- Many code lines can be on the same line separated by semicolon
- One code line can go over many lines until a semicolon is found
- Elements of the same line are separated by white spaces



BLOCKS

- Block of lines, e.g. a function body, is enclosed between braces {...}

```
int main()
{
    double r = 10;
    double area = PI * r * r;
    cout << area;
    return 0;
}
```



COMMENTS

- Comments are ignored by the compiler
- Comments are typically used by programmer to document code

- Single-line comments start with `//` and ends at the end of line

```
// This is a whole-line comment
```

```
double area = PI * r * r; // This is an end-of-line comment
```

- Multiple-line comments starts with `/*` and ends with `*/`

```
/* This is  
a multi-line  
comment */
```



VARIABLES

- Variables in C++ must be defined in advance
- Each variable must have a name and a data type
- Available data types include int, char, float, double, ...

```
double r; // Define a variable named r of type double
int x;    // Define a variable named x of type integer
```



INPUT & OUTPUT

```
#include <iostream>
#define PI 3.14
using namespace std;

int main()
{
    double r;
    cin >> r;
    double area = PI * r * r;
    cout << area << endl;
    return 0;
}
```



SUMMARY

- Program Structure
- Program Execution
- Preprocessor Directives
- Lines & Blocks
- Comments
- Variables
- Input & Output



QUIZ

Write a program to calculate and print the area of a rectangle



ASSIGNMENT 0

1. Create a **GitHub** account (<https://github.com>)
2. Go to <https://classroom.github.com/a/TldKtTY6>
3. Clone project to your computer
4. Edit ***solution.cpp*** in IDE
5. Make sure it complies and runs with no errors
6. *Add, commit, and push* to Github
7. Copy Github project link
8. Submit link on course website as solution for assignment 1



NON-ACADEMIC READING

