INTRODUCTION

Compilers, IDEs, Hello World

Florian Warg, Max Staff

April 26, 2017

COMPILERS

Name	OS	License	C++11	C++14	C++17
Intel	Most	Proprietary	Partial	Partial	No
MS Visual C++	Most	Proprietary	Yes	Partial	Partial
Clang	All	MIT-like	Yes	Yes	Partial
GCC	All[1]	GPLv3	Yes	Yes	Partial

Source

[1] Using external software for Windows

IDES

Name	OS	License	Autocomplete	Tools
Visual Studio	Windows	Proprietary	Complex	Most[2]
Eclipse CDT	Java	EPL	Complex	Most[2]
Xcode	OS X	Proprietary	Complex	Most[2]
CLion	Java	Proprietary	Complex	Many[3]
Qt Creator	All	LGPL	Complex	Many[4]
Geany	All	GPL	Simple	Class Browser

Source

[2] GUI builder, class browser, toolchain, debugger, profiler, refactoring, code coverage
[3] Except for code coverage, GUI builder, profiler
[4] Except for code coverage

HELLO WORLD

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

int main() {
    cout << "Hello World!\n";
    return 0;
}
```

Compile with:

```
g++ -std=c++14 -Wall -Wextra hello_world.cpp
```

READING INPUT

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  int main() {
6      cout << "Please enter your name: ";
7      string name;
8      cin >> name;
9      cout << "Hello " << name << "!\n";
10     return 0;
11 }
```

FORMATTING OUTPUT

```
1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  using namespace std;
5
6  int main() {
7      cout << "Please enter the height and width:\n";
8      int height, width;
9      cin >> height >> width;
10     cout << "Height: " << setw(4) << height << "\n";
11     cout << "Width:  " << setw(4) << width << "\n";
12     cout << "Area:   " << setw(4)
13          << height * width << "\n";
14     return 0;
15 }
```

DATA TYPES

```
1  // string
2  string a = "abcde";
3  char b = a[4];
4  char c;
5  c = 'e';
6
7  // integer
8  int d = 42;
9  int e, f = 12;
10 // what's the value of e?
11
12 // float
13 float g = 13.37;
```

MORE DATA TYPES

```
1  // similar to int
2  short f = 5;
3  long g = 5812374593124583245 8L;
4
5  // similar to double
6  double h = 123456789.123456789;
7
8  // int again...
9  unsigned int i = 1337U;
10
11 // and to interpret conditions
12 bool a = false;
13 bool b;
14 b = !a;
```

OPERATIONS - INTEGER

```
1  // basic maths
2  int a = 12,
3      b = 15;
4
5  int c = a + b;
6  c += b;
7
8  d = a - b;
9  d -= a * b;
10 e *= d;
11
12 f = a + b / b;
13 f /= c;
```

OPERATIONS - INTEGER

```
1  // basic programming tools
2  int a = 12,
3      b = 15;
4
5  int c = b % a;
6  int d = a++;
7  int e = ++a;
8
9  --d;
10 e--;
```

OPERATIONS - INTEGER

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

int main() {
    long int a = -5, b = 16;
    unsigned int c = a;
    long unsigned int d = a;
    short int e = c;
    long int f = e, g = c;
    cout << "a: " << a << "\n" "b: " << b << "\n";
    cout << "c: " << c << "\n" "d: " << d << "\n";
    cout << "e: " << e << "\n" "f: " << f << "\n";
    cout << "g: " << g << "\n";
    return 0;
}
```

OPERATIONS - STRINGS AND CHARACTERS

```
1  // concatenation
2  string a = "Hello ",
3       b = "World!";
4  string c = a + b;
5  string d = "Hello" "World!";
6
7  // characters
8  char e = 'e';
9  char f = e + 1;
```

OPERATIONS - COMPARISON

```
1  bool a = true, b = false;
2  bool c = a == b;
3
4  if (a != b) {
5      bool d = a > b;
6      d = !(a <= b);
7  }
8
9  bool e = a && b;
10 bool f = a || b;
```

OPERATIONS - BITWISE

```
1  int a = 12, b = 6;
2
3  int c = a & b;
4  int d = a | b;
5
6  int e = ~a;
7
8  int f = a ^ b;
9
10 int g = a << b;
11 int h = a >> b;
```

COMPOUND OPERATORS

```
1  int a = 12, b = 6;
2
3  a += b; a += b;
4  a *= b; a /= b;
5
6  a %= b;
7
8  a &= b;
9  a |= b;
10
11 a ^= b;
12
13 a <<= b;
14 a >>= b;
```

OPERATOR PRECEDENCE

LOOPS

```
1  int amount = 5;
2  string text = "test";
3
4  while (amount > 0) {
5      cout << text << "\n";
6      --amount;
7  }
8
9  for (int i = 0; i < amount; ++i) {
10     cout << text << "\n";
11 }
12
13 for (auto i : text) {
14     cout << i << "\n";
15 }
```

LOOPS

```
1  int amount = 200;
2  string text = "test";
3  string longer_text = "";
4
5  while (amount > 0) {
6      longer_text += text;
7      if (longer_text.size() > 100) {
8          continue;
9      }
10     if (longer_text.size() > 120) {
11         break;
12     }
13 }
```

FUNCTIONS

```
1  void myPrint(string parameter) {
2      cout << parameter << "\n";
3  }
4
5  int triangleArea(int sideA, int sideB) {
6      return sideA * sideB / 2;
7  }
8
9  int faculty(int nr) {
10     if (nr > 0) {
11         return nr * faculty(nr - 1);
12     } else {
13         return 0;
14     }
15 }
```