

ENGG1340

C++ Notes for HKU · Spring 2024

Author: Jax

Contact: enhanjax@connect.hku.hk

MORE notes on my [website!](#)

1 C++ basics

Basic differences to Python:

1. C++ is a *compiled* language, which means that the code needs to be compiled before it can be run.
2. C++ is a *statically typed* language, which means that the type of a variable is known at compile time.
3. C++ expressions end with a semicolon `;`.

1.1 Initialization

For a `.cpp` file, we must include the following contents:

```
// include different libraries
#include <iostream> // iostream provides basic io functions
using namespace std; // use the "std" namespace
int main()
{
    // code to be ran when program starts
}
```

1.2 Variables and IO

1.2.1 Variables types

```
char 'b' // only one character
int 1 // integer
float 1.0 // floating point number
double 1.0 // double precision floating point number
bool true // boolean

const int a = 1; // constant variable that cannot be changed
```

Note that there are no built-in string types in C++. A string is simple an **array of characters**.

1.2.2 Defining variables

```
int a = 1, b = 2, c = 3;
double a, b, c; // empty initialization
```

1.2.3 Input and output

```
cin >> name >> age; // read multiple inputs to variables
cout << "Hello " << name << endl; // Hello <name>
```

1. `>>` is the *extraction operator*
2. `<<` is the *insertion operator*
3. We can use `\` to escape special characters
4. `endl` is a special character that represents a new line, equiv to `\n`

1.2.4 Decimal points

We can fix the number of decimal points by using the `fixed` and `setprecision` functions from the `iomanip` library.

```
#include <iomanip>                // include iomanip library
cout << fixed << setprecision(2); // set precision to 2 decimal places
cout << 3.14159;                 // 3.14
```

1.3 Operators

```
n = ++i;    // equiv to i = i + 1, n = i;
n = i++;    // equiv to n = i, i = i + 1;
n = 1 && 0;  // logical and
n = 1 || 0;  // logical or
n = !3;     // logical not, converts any number > 0 to 0 (!3 => 0)

// C++ contains the usual operators: +, -, *, /, %
10 / 3 // If both types are integers, the result is an integer (3)
10.0 / 3 // If either type is a float, the result is a float (3.3333)
```

1.4 Control flow

1.4.1 Basic conditionals

```
if (a == 1) {
    // do something
} else if (a == 2) {
    // do something else
} else {
    // do something else
}
```

1.4.2 Switches

```
switch (ctrl_expr) {
    case const_1: // if ctrl_expr == const_1
```

```

        break; // man
    case const_2: // if ctrl_expr != const_1 && ctrl_expr == const_2
        break;
    default: // if no case matches (optional)
        break;
}

```

1.4.3 Ternary operators

```

(condition ? if_true : else_false) // returns the corresponding expression

```

1.4.4 Loops

```

for (int i = 0; i < 10; i++) { // initialize, check conditions, increment
    // do something 10 times
    continue; // skip the rest of the loop
    break; // break out of the loop
}

while (a < 10) { // check conditions first
    // do something
}

do { // execute first
    // do something
} while (a < 10); //then check conditions

```

1.5 Pass by reference

Pass-by-reference means to pass the **reference of a variable** as the parameter of a function. The called function can **modify the value of the variable** by using its reference passed in.

```

void swapnum(int &i, int &j) { // pass by reference
    int temp = i;
    i = j;
    j = temp;
}

int main(void) {
    int a = 10;
    int b = 20;
}

```

```
    swapnum(a, b);                                // a = 20, b = 10
    return 0;
}
```

1.6 Functions

```
//return_type func_name (type var1, type var2)
double func_name (int a, int b) {
    return a + b;
}
```

Note that nested function declaration is not allowed.

1.7 Arrays

```
int a[10]; // array of 10 integers
int a[] = {1, 2, 3, 4, 5}; // array of 5 integers
int a[10][10]; // 2D array of 10x10 integers
char a[] = "Hello"; // array of 5 characters
```

You can also use the string library to work with strings:

```
#include <string>
string a = "Hello"; // string class
```