

Selection Statements & Logical Operators

Tai, Wei Hsuan

week 3



1. Recap
2. Force data type casting
3. Selection Statements
4. Logical Operators

Recap

Remember

- Every statement ends with a semicolon (;).
- Programs always start from the `main()` function.
- All functional code needs to be inside a function.
- All functions must be enclosed in curly braces {}.

Basic structure of C++

```
1      #include <bits/stdc++.h>
2      using namespace std;
3
4      int main() {
5          // code here
6          return 0;
7      }
```

Variables Declaration

```
1      int a;  
2      double b, pi=3.14;  
3      char c='A';
```

Basic Input/Output

```
1      int a, b, c;  
2      cin >>a>>b>>c;  
3      cout<<"The result of a*10 is"<<a*10<<endl;
```

What's the value of each variable?

```
1      int a, b;  
2      char c;  
3      //input: 10+5  
4      cin >>a>>c>>b;
```

A little tip

The result is `a=10`, `b=5`, `c='+'`

In C++, the input stream will read the whole input until it meets a whitespace, newline or another data type.

Therefore, `cin` will read 10 into `a`, then meet the character `'+'` and stop reading for `a`.

Next, `cin` will read `'+'` into `c`, then meet the character `'5'` and stop reading for `c`.

Finally, `cin` will read 5 into `b`.

This is a little tip for you to understand how `cin` works.

Force data type casting

Idea of force data type casting

In C++, we should declare a variable with data type before we utilize it. But what if we assign a value with different data type to it?

In this case, C++ will automatically convert the assigned value to the data type of the variable.

For instance, we declare an integer variable `a`, but we assign a double value `3.14` to it.

In this case, C++ will automatically convert `3.14` to `3` and assign it to `a`.

Conversion rules

- Integer to double: add a decimal point and a zero.
- Double to integer: remove the decimal point and the digits after it.
- Character to integer: convert the character to its ASCII code.
- Integer to character: convert the ASCII code to its corresponding character.

Little Practice

We give all letters a score according to their position in the alphabet.

For example, $a=1$, $b=2$, $c=3$, ..., $z=26$.

Given a lowercase letter, tell me its score.

Hint: you may need to use force data type casting.

Solution

We know that all letters have their own ASCII code, and the ASCII code is continuous.

Therefore, we can just calculate the distance between the given letter and 'a'.

Remember to add 1 to the result, because 'a' is the first letter.

```
1      char c;  
2      cin >>c;  
3      cout<<(int)(c-'a'+1)<<'\n';
```

You can add brackets to fix the target data type.

Selection Statements

Preface: How to convert score to GPA?

Following is the GPA conversion table of an university.

Score	Grade	GPA
80-100	A	4.0
70-79	B	2.8
60-69	C	1.5
0-59	F	0.0

Table 1: Grade Conversion Table

Solution without selection statements

With only variable operations, we can do it like this:

```
1      #include<bits/stdc++.h>
2      using namespace std;
3
4      int main(){
5          int score;
6          double gpa;
7          cin >>score;
8          gpa=(score>59)*1.5+(score>69)*1.3+(score>79)*1.2;
9          cout<<gpa<<'\n';
10     }
```

But this notation is not easy to read. We have a more readable way to do it.

Overview of Selection Statements

Selection statements are used to select a block of code to be executed based on a condition.

In C++, we have three selection statements: `if`, `if-else` and `switch`. In fact, `switch` is not commonly used, so we will focus on `if` and `if-else` today.

Structure of Selection Statements

```
1      if(condition) {  
2          // code to be executed if condition is true  
3      }else if(condition2){  
4          // code to be executed if condition2 is true  
5      }else{  
6          // code to be executed if both condition and  
           ↪ condition2 are false  
7      }
```

Note that condition must be a boolean expression, which means it can only be true or false. In C++, any non-zero value is considered as true, and 0 is considered as false.

We often use a logical expression as the condition.

Back to GPA conversion

```
1      #include<bits/stdc++.h>
2      using namespace std;
3
4      int main(){
5          int score;
6          double gpa=0;
7          cin >>score;
8          if(score>79){
9              gpa=4.0;
10         }else if(score>69){
11             gpa=2.8;
12         }else if(score>59){
13             gpa=1.5;
14         }
15         cout<<gpa<<'\\n';
16     }
```

Given a letter, tell me it is uppercase or lowercase.

Hint: Remember that letters are continuous in ASCII table.

Solution

```
1      #include<bits/stdc++.h>
2      using namespace std;
3
4      int main(){
5          char c;
6          cin >> c;
7          if(c>='A' && c<='Z'){
8              cout<<"Uppercase"<<endl;
9          }else if(c>='a' && c<='z'){
10             cout<<"Lowercase"<<endl;
11         }
12     }
```

Logical Operators

Overview of Logical Operators

In Selection Statements, logical expressions are often used as the condition.

In fact, computers would calculate the logical expressions to determine whether the condition is true or false, then return the result to the selection statements.

Logical Operators

In C++, we have three logical operators: AND (`&&`), OR (`||`) and NOT (`!`), and we've learned them in the previous course. Let's take a recap.

A	B	A && B	A B	!A
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

Table 2: Truth Table of Logical Operators

With logical operators, we can combine multiple conditions into one condition.

For example, we want to check if a number is between 1 and 10.

We can do it like this:

```
1      int a;  
2      cin >>a;  
3      if(a>=1 && a<=10){  
4          cout<<"a is between 1 and 10"<<endl;  
5      }
```

Given two integers, tell me how many even numbers are between them (inclusive). For example, given 3 and 9, the even numbers between them are 4, 6, 8.

In this problem, loop is forbidden because the number of integers between the two given integers may be too large.(e.g. 10^{15})

There are 3 ants on a stick of length L . The positions of the ants are a , b and c . Each ant will randomly move to the left or right with equal probability. When two ants meet, they will turn back and move in the opposite direction. When an ant reaches the end of the stick, it will fall off. Calculate the expected time for all ants to fall off the stick.

There's a robot on a 2D plane. The robot starts at the origin $(0, 0)$ and faces the positive x-axis. At t^{th} second, the robot will move forward t units, then turn left 90 degrees. After n seconds, where will the robot be?