

UEE1302(1066) F12: Introduction to Computers and Programming

Flow of Control (II) - Repetition



What you will learn from Lab 4

In this laboratory, you will understand how to use repetition (for and while) to control the flow of programs.

TASK 4-1 : FLOW CONTROL STATEMENT - for

- ✓ Please predict the result of program lab4-1, and then execute this program to compare the result on screen with what you predict.

```
//File: lab4-1.cpp
#include <iostream>
using namespace std;

int main()
{
    cout << "Open the refrigerator..." << endl;
    int count = 0;
    for (int idx = 0; idx < 10; idx+=2)
    {
        cout << "Counter in loop " << count++ << "\t";
        cout << "Number of object: " << idx << endl;
    }
    cout << "...close the refrigerator." << endl;

    return 0;
}
```

- ✓ An example for multiple for structure

```
//File: lab4-2.cpp
#include <iostream>
using namespace std;

int main()
{
    for(int i=1;i<=10;i++)
    {
```

```
        for(int j=1;j<=i;j++)
            cout<<j<<" ";
        cout<<endl;
    }

    return 0;
}
```

TASK 4-2 : FLOW CONTROL STATEMENT - while

- ✓ Program lab4-3 is designed to print out the values from 1 to 10. Execute the program below and observe the results. If the results are different from the expected, please modify the program properly.

```
//File: lab4-3.cpp
#include <iostream>
using namespace std;

int main()
{
    int i;
    i = 1;
    while (i < 10)
    {
        cout << i << endl;
        i++;
    }

    return 0;
}
```

- If “i++” is removed from the above program, what can you conclude for the result?
 - ✧ Note: “**Ctrl+C**” is the way to terminate the running program.
- Rewrite File lab4-3.cpp by replacing ‘while’ with ‘for’

- ✓ Execute program lab 4-4 and record your result.

```
//File: lab4-4.cpp
#include <iostream>
using namespace std;

int main()
{
```

```
int reachcont = 1;
while (reachcont)
{
    char sym;
    cout << "Enter a symbol (a, b, or c): ";
    cin >> sym;
    switch (sym)
    {
        case 'a':
        case 'b':
        case 'c':
            cout << "The symbol is " << sym << endl;
            reachcont = 0;
            break;
        default:
            cout << "The symbol is not a, b, or c." << endl;
            break;
    }

    return 0;
}
```

- ✓ Execute program lab4-5 and record your result.

```
//File: lab4-5.cpp
#include <iostream>
using namespace std;

int main()
{
    int i;
    i = 10;
    do
    {
        cout << i << endl;
        i++;
    }while (i<10)

    return 0;
}
```

- Please modify the program by replacing “do ... while” structure with “while”.

➤ Note that the ; (semicolon) is required in “do ... while” structure.

✓ Execute the program lab4-6 and record your result.

```
//File: lab4-6.cpp
#include <iostream>
using namespace std;

int main()
{
    int i;
    i = 10;
    while (i < 10)
    {
        cout << i << endl;
        i++;
    }

    return 0;
}
```

➤ Note: Observe carefully the differences between results from two programs using “while” (lab4-6.cpp) and ”do...while”(lab4-5.cpp) structures.

TASK 4-3 : EXERCISES

- ✓ Write a C++ program to print out the multiplication table.

```
> ./ex4-1
Enter the size of multiplication table(1~9): 3
1 * 1 = 1      1 * 2 = 2      1 * 3 = 3
2 * 1 = 2      2 * 2 = 4      2 * 3 = 6
3 * 1 = 3      3 * 2 = 6      3 * 3 = 9
>
```

```
> ./ex4-1
Enter the size of multiplication table(1~9): 10
The size of multiplication table should be 1~9.
Enter the size of multiplication table(1~9): 2
1 * 1 = 1      1 * 2 = 2
2 * 1 = 2      2 * 2 = 4
>
```

- ✓ Write a C++ program to print out all prime numbers which is smaller than and equal to the input number. Please meet the requirements as follows.

```
> ./ex4-2
Enter an integer value:
28
All prime numbers below 28 are:
2 3 5 7 11 13 17 19 23
>
```