SLIIT

Discover Your Future

IT1050- Object Oriented Concepts

Lecture – 02 - C++

Learning Outcomes

At the end of the Lecture students should be able to
Write a C++ program including :

- Namespaces
- Variables
- Sequence
- Selection
- Repetition
- Using Input Commands and Formatting Output

The `std` namespace

- Let's have a look at the iostream.h header file (a simplistic view of the actual file)

```
// iostream.h header file
// this is inserted to your program when you use the command #include <iostream>

namespace std {

    // various commands related to input and output are defined here

    ofstream cout;

    //
    ifstream cin;

    char endl = '\n';
}
```

To access cout, cin, endl outside the namespace we have to explicitly use

std::cout
std::cin
std::endl

Everything defined in the iostream header file is defined under a namespace called std

namespaces are used to avoid naming collisions

```
// I could write my code for example
// using the namespace FOCSLIIT

// Imagine a namespace to be a folder
// in your computer.
namespace FOCSLIIT {
      int data;
      void graphics(int x, int y);
}
namespace Graphics {
     void graphics(int x, int y);
     int mygraphics;
     int data;
}
// Since FOCSLIIT and Graphics are
// two separate namespaces (folders)
// variables, functions with the same
// name can exist without issues
```

```
FOCSLIIT::data
FOCSLIIT::graphics(10,20);


Graphics::data
Graphics::graphics(10,20);
```

namespace1.cpp

namespace2.cpp

The `std` namespace

- The `std :: ` before `cout` is required when we use names that we've brought into the program by the preprocessing directives `#include <iostream>`
- Means `cout` belongs to namespace `std`

```
// C++ Program
#include <iostream>

int main ( )
{
   std::cout<< "Hello ";
   std::cout<<"World !";
   std::cout << std::endl;

   return 0;
}
```

```
// C++ Program
#include <iostream>
using namespace std;

int main ( )
{
   cout<< "Hello ";
   cout<<"World !";
   cout << endl;

   return 0;
}
```

namespace std

using namespace std;

- The keyword `namespace` defines a scope
- `std` is a namespace defined by C++
- `cout` is included in `std`
- Using the above statement will omit having to use `std::` (`::` - scope resolution operator) with every member (directive/keyword) of the `std` namespace

Chaining multiple << operators together

- Instead of using only one << for each cout. We can chain multiple insertion operators in the same line. Each of the data can be of different types.

```
// C++ Program
#include <iostream>
using namespace std;

int main ( )
{
   cout << " My Score is  ";
   cout << 70;
   cout << endl;

   return 0;
}
```

```
// C++ Program
#include <iostream>
using namespace std;

int main ( )
{
   cout << "My Score is " << 70 << endl;
   return 0;
}
```

C++ Keywords

Table 4 — Keywords

alignas	continue	friend	register	true
alignof	decltype	goto	reinterpret_cast	try
asm	default	if	return	typedef
auto	delete	inline	short	typeid
bool	do	int	signed	typename
break	double	long	sizeof	union
case	dynamic_cast	mutable	static	unsigned
catch	else	namespace	static_assert	using
char	enum	new	static_cast	virtual
char16_t	explicit	noexcept	struct	void
char32_t	export	nullptr	switch	volatile
class	extern	operator	template	wchar_t
const	false	private	this	while
constexpr	float	protected	thread_local	
const_cast	for	public	throw	

Use of Variables

- Same as in C

```
// prg_02.cpp
//Program that adds two numbers
#include <iostream>
int main ( )
{
    int number1 = 25;
    int number2 = 32;
    int sum;
    sum = number1 + number2;
    cout<< "Sum is : " << sum << endl;  // Display value of
sum
    return 0;
}
```

Recall…….

- C++ Rules for making identifies
  - Consists with letters, digits, and underscore character
  - Starts with a letter
  - Cannot contain spaces, special characters, operators and reserve words / keywords
  - Cannot contain more than 31 characters

Input from Keyboard

```c
// C Program
#include <stdio.h>

void main ( void)
{
   int num;
   printf (“Input Number :  ”);
   scanf(“%d”, &num);
   printf(“Number : %d\n”, num);

}
```

```cpp
// C++ Program
#include <iostream>
using namespace std;

int main ( )
{
    int num;
    cout<< “Input Number :”;
    cin >> num;
    cout<<“Number is : “<< num
                          <<endl;

   return 0;
}
```

`cin` command

- `cin` is the stream input in the C++ standard library.
- The Extraction operator >> will skip leading whitespace

  (blank, tab, newline) characters and start the value with the first non-whitespace characters.
- A value is terminated by whitespace.
- e.g:

```
cout<< “Input length and width :”;
cin >> length >> width;
```

Output

```
Input length and width : 7.5     8.5
```

Exercise 01

- Write a C++ program to input the length and the width of a rectangle and calculate and print the perimeter.

Try this in **repl.it** using your account. Copy your solution's url to chat window

Formatting Output – <iomanip>

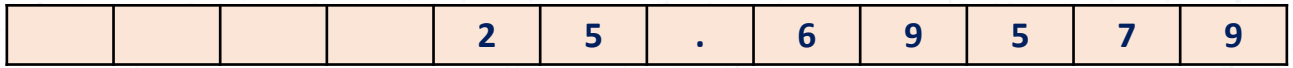
- These allow you to send control signals to cout to control how the output is displayed.
- **`setw(n)`** – Specifies number of spaces used to display a number
- **`setiosflags (ios::fixed)`** – Specifies that the number should be printed as floating point number with decimal places e.g. 345.67
- **`setprecision (n)`** – When used with ios::fixed, controls the number of decimal places that will be printed.

Formatting Output cont....

```
cout<< setw(12) <<setiosflags(ios::fixed) << 25.695789<< endl;
```


```
cout<< setw(12) <<setprecision (3) << 25.695789<< endl;
```


```
cout<< setw(12) << setprecision (5) << 2 5.695789 << endl;
```



iomanip.cpp

Exercise 02

- Modify the program that was written to calculate the perimeter of the rectangle to display the results using two decimal places.

Try this in **repl.it** using your account. Copy your solution's url to Slido.com

Selection Control Structure

- if
- if- else
- switch

- ```
  if( a >b )
  {
      cout <<a <<“is the largest”<<endl;
  }
  ```

- ```
  if ( a > b )
      cout<< a << “is greater than”<< b<<endl;
  else
      cout<<b <<“is greater than”<<a<< endl;
  ```

Selection Control Structure

```
if( score == 4 )
   cout <<“Excellent”<<endl;
else
   if ( score == 3 )
      cout<<“Good”<<endl;
   else
      if ( score == 2)
           cout<<“Average”<<endl;
     else
      {
       cout<<“Below Average”<<endl;
       cout<<“Needs Improvement”;
      }
```

```
switch( score )
{
 case 4 :cout <<“Excellent”<<endl;
            break;
 case 3 :cout<<“Good”<<endl;
            break;
 case 2 :cout<<“Average”<<endl;
            break;
 default :
           cout<<“Below Average”<<endl;
            cout<<“Needs Improvement”;
 }
```

Exercise 03

- Write a C++ program to input the total price to be paid by a customer and calculate the discount according to the chart below.

Total Price	Discount Rate
> 10000	25%
10000 - 5000	15%
5000 - 3000	10%

Try this in **repl.it** using your account. Copy your solution's url to chat window

Iteration Control Structure

- while
- do – while
- for

```
int count=1;
while ( count <=10 )
{
    cout<<count<<endl;
    count ++;
}
```

```
int count=1;
do
{
    cout<<count<<endl;
    count ++;
} while ( count <=10 );
```

```
for(int count=1; count <=10 ; count ++)
        cout<<count<<endl;
```

Exercise 04

- Display number 1000,900,800,700,… 100

Using a while loop, do while loop and a for loop within the same program.

i.e. Display these number series three times. One for each repetition structure.

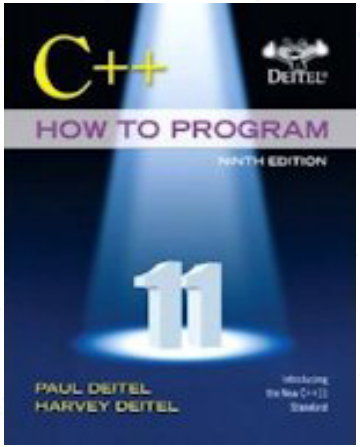
Try this in **repl.it** using your account. Copy your solution's url to chat window

Exercise 05

- Consider Exercise 03
- Modify the program to input details of 3 customers and calculate the total discount amount given.
- What would you do if you want to continue entering prices until -1 is entered ?
- What would you do if you want to enter data until user enters ‘y’ to continue and ‘n’ to stop?

Try this in **repl.it** using your account. Copy your solution’s url to chat window

Reference



Chapter 01 & 02

Deitel & Deitel's (2016), C++ How to Program, 9th Edition


THANK
YOU