SLIIT

Discover Your Future

IT1050- Object Oriented Concepts

Lecture-01 – Introduction to C++

SLIIT
FACULTY OF COMPUTING

Agenda

- Introduction to Module
- Recalling C
- C to C++

Introduction to Module

Learning Outcomes

- Understand and apply the basic concepts of Object Oriented Programming
- Design solutions by identifying the classes and relationships (Object Oriented Analysis and Design)
- Implement a solution to the given problem using the C++ Language

Delivery

- Lectures
  - 1 Hour per week
- Tutorial
  - 1 Hour per week
- Labs
  - 2 Hours per week

Assessment Criteria

- Continuous Assessment
  - Assignment 1 - Practical 10%
  - Assignment 2 - Group work (case study) 10%
  - Mid Term Examination 20%
- Final Examination 60%

Content

- **Introduction to C++**
- **Introduction to OOP Concepts**
  - Abstraction
  - Encapsulation
  - Information Hiding
- **Identifying classes and objects**
- **Object Oriented Design**
  - Noun Verb Analysis
  - CRC Cards
- **Introduction to Object Oriented Programming**
- **Advanced Object Oriented Concepts**
  - Relationships
  - Polymorphism

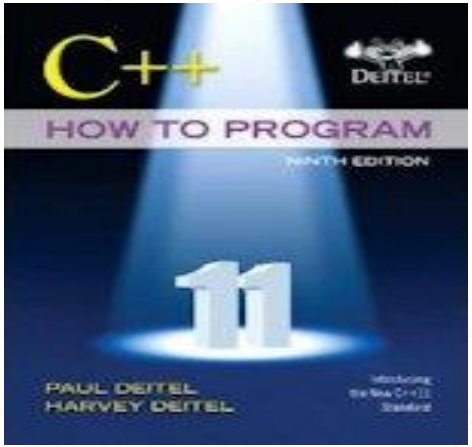
Teaching Learning Activities

- Case Study - Library System
- Home work - Watching Videos
- Group work - Assignment 2

Academic Integrity Policy

- Are you aware that following are not accepted in SLIIT???
  - Plagiarism - using work and ideas of other individuals intentionally or unintentionally
  - Collusion - preparing individual assignments together and submitting similar work for assessment.
  - Cheating - obtaining or giving assistance during the course of an examination or assessment without approval
  - Falsification – providing fabricated information or making use of such materials
- From year 2018 the committing above offenses come with serious consequences !
- See General support section of Courseweb for full information.

Reference

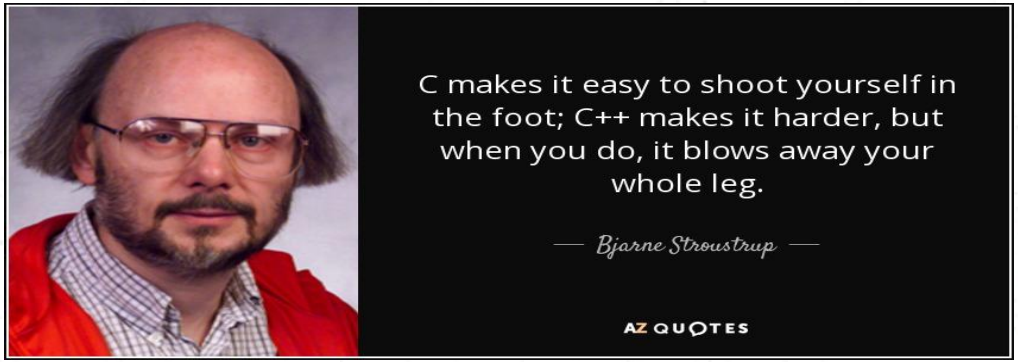


Deitel & Deitel's (2016), C++ How to Program, 9th Edition

Grady Brooch (2008), Object-Oriented Analysis and Design with Application, 3rd Edition

C++

- One of the most powerful and popular programming languages
- Evolve from C
- Developed by Bjarne Stroustrup in 1979 at Bell Laboratories
- Provide capabilities for Object Oriented Programming
- Current Version – C++ 17



C vs C++

```c
// C Program

#include <stdio.h>

void main ( void)
{

   printf ("Hello World  ! \n");

}
```

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

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

   return 0;
}
```

Output :

```
Hello World !
```

First C++ Program

```
// C++ Program  :  prg_01.cpp
//Printing a String
#include <iostream>  // allows program to output data to the screen

int main ( )   // Function main begins program execution
{
     std::cout<< “Hello World !“; // Display message
     std::cout<< std::endl;  // New line

     return 0;   // indicate that program ended successfully

}  // End of main function
```

Comments

```
// C++ Program  :  prg_01.cpp
//Printing a String
```

- Comments provide information to the people who read the program
- Comments are removed by the preprocessor, therefore the compiler ignores them
- In C++, there are two types of comments
  - Single line comments //
  - Delimited comments /* */ for comments with more than one line.

Preprocessing Directives

#include <iostream>

- Lines begin with # are processed by the preprocessor before the program is compiled.
- Notifies the preprocessor to include in the program the content of the input/output stream header <iostream>
- “iostream” is a header file containing information used by the compiler when compiling a program with output data to screen or input data from the keyboard using c++ input/output stream

The `main` function

```
int main()
{

}
```

- C++ programs begin executing at function `main`.
- It is the main building block of a program.
- `int` indicates that main returns an integer value.
- `{` (left brace) indicates the begin of the main body and `}` (right brace) indicates the end of the function's body.

Output Statement

```
std:: cout<< “Hello World ! “ ;
```

- **`cout`** : to indicate the computer to output something on screen
- **`<<`** : is the stream insertion operator used to send information to `cout`
- **`“Hello World !”`** : String / String Literal. What you need to display on screen
- **`;`** : statement terminator

New Line

```
std:: cout<< endl ;
```

- **`endl`** : to go to a new line (same as “\n”)

  eg : std::cout<<“\n”;

```
std:: cout<< “Hello World !”<< endl ;
```

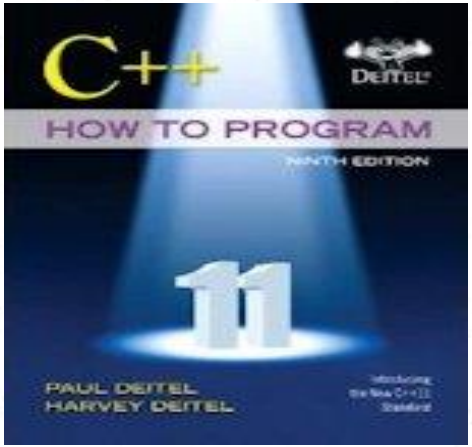
Output :

```
Hello World !
```

Exercise

- Write a C++ program to display your name and address in 3 lines.

Reference



Chapter 01 & 02

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