Programming Fundamentals (COMP1112) Lecture 1

Division of Science & Technology

University of Education, Lahore.

Objectives

- Learn the concepts of programming using a standard language.
- Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
- Develop abilities to write a computer program to solve specified problems.
- Use the selected programming language to create, debug and run programs.

Course contents

- **Introduction to Programming:** Problem solving, Introduction to programming and programs, Types of languages, Compiler vs. Interpreter.
- **Overview of Programming:** Structure of Program, indentation and coding conventions, Program output, Syntax rules, Common errors, Language keywords.
- **Data Types:** Identifiers, Constants, Types of variables, Type conversion and casting, Type promotion rules.
- **Operators:** Arithmetic operators and punctuations, Precedence, Associations, Equality and relational operators.
- **Control Statements:** Selection structure (if/, if/else), Multiple selection structure (switch and break), break and continue, Loops.
- **Arrays:** Declaring arrays; Initialization, Multidimensional arrays, Two dimensional arrays, Example (Matrix manipulation), Assignments, and Mini Project.
- **Methods/Functions:** Library functions, Modular Approach, Functions, and Function definitions. Function prototypes.
- **Class Fundamentals:** Class syntax, General form of class, Object declaration.

Books

- C++ How to Program
  By Deitel & Deitel

- The C++ Programming Language
  By Bjarne Stroustrup

- Object-Oriented Software Engineering
  By Jacobson, Christerson, Jonsson, Overgaard

Grading policy

- Practical 15 %
- Sessional 20 %
- Mid-Term 20 %
- Final 45 %

Program

- A set of instructions written in some programming language to solve a particular problem. Goal is optimization (minimum number of lines and system resources)
- Programming is the process of solving a problem with the help of computer system. It prepares different instructions for computer.

Program Development Process

- Defining and analyzing problem
- Designing the algorithm
- Coding or writing a program
- Testing program

Integrated Development Environment (IDE)

IDE contains

- Editor - use to write programs
- Compiler - convert source code into object code and display syntax errors
- Debugger- use to find errors
- Linker - combines object program with additional libraries and save as .exe file
- Loader - places executable file into memory

Types of languages

- High level languages
  - Procedural languages
    - execution is in same sequence
    - Program in pre-defined set of instructions
    - FORTRAN (Formula Translation)
    - COBOL (Common business oriented language)
  - Object-oriented languages
    - Objects
    - Real world modeling
    - Reusability
    - Modularity
    - C++, Java
  - Non-procedural languages
    - User only needs to tell the computer what to do
    - SQL (structured Query Language)
- Low level languages
  - Machine language - Instructors are in binary form
  - Assemble language - one step higher than machine language

History of C++

- Started developing in 1980
- First commercial release in 1985
- Originally called “ C with classes”
- Allows the use of Object oriented programming technique

Features of C++

- Build on OOP paradigm
- Case sensitive
- Reusability
- Modularity

Language processors

- Compiler - converts programs into machine code as a whole and identifies syntax errors
- Interpreter - converts one statement of program one time and does not produce object code
- Assembler - translates assembly language instructions into machine language

Basic structure of C++ program

- Preprocessor directive
- main() function
- Program body (C++ statements)

Pre-processor directive

- Instruction given to compiler before the execution of program
- Processed by program known as preprocessor
- Part of C++ compiler
- Modified C++ source code before compilation
- Starts with symbol #
- Preprocessor directive in C++ to include header files

#include<iostream>

The above statement tell the compiler to include the file iostream.h in source code before compiling it.

main () function

- Execution of C++ program starts from main()
- Each program must contains a main() function
- Syntax of main()

```
main ()
{
// body of main
}
```

First C++ program

```
#include<iostream>
using namespace std;
main()
{
cout<<“Hello World!”;      //<< insertion operators


}
```

Error types

- Syntax
- Logical
- Run-time

References

- C++ How to Program
  By Deitel & Deitel

- The C++ Programming Language
  By Bjarne Stroustrup

- Object-Oriented Software Engineering
  By Jacobson, Christerson, Jonsson, Overgaard

- Object oriented programming using C++ by Tasleem Mustafa, Imran Saeed, Tariq Mehmood, Ahsan Raza