Lab # 1: Introduction to Programming
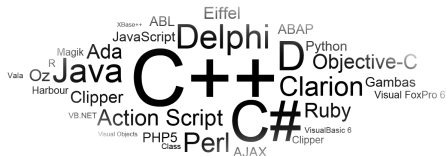## EC-102 – Computer Systems and Programming

Usman Ayub Sheikh

School of Mechanical and Manufacturing Engineering (SMME),
National University of Sciences and Technology (NUST)

September 10, 2015

Outline

Lab Grading Criteria

- Lab Work and Lab Report $\sim 50\%$
  - Lab work represents your performance in lab assignments/tasks. Every student will be graded individually.
  - Lab report may be submitted by a group of max 5 students. Make sure that you submit one lab report per week.
- Projects $\sim 40\%$ (details will be provided later)
- Attendance $\sim 10\%$
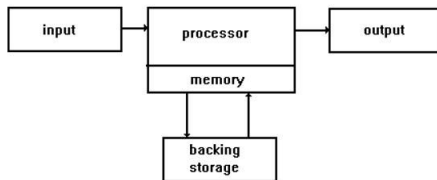
Lab Report: Contents

- Problem statement
- Algorithm
- Flow chart
- Code
- Conclusion

Lab Report: Title Page

- Name of school and university
- Name of subject
- Lab number and topic
- Submitted to
- Submitted by
- Date

Basics of Computer Programming

A **computer system** is one that is able to take a set of inputs, process them and create a set of useful outputs.


One or more *input*s are used to provide data, this data is then *processed in some way* and the outcome of processing is sent to an *output* or it may be stored until some event happens and brings it to the output.

For processing to take place, there needs to be a set of instructions of what needs to be done. This set of instructions is known as a **computer program**.

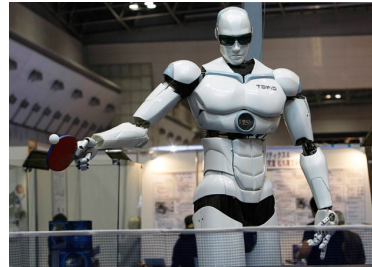- Automation – home, industrial
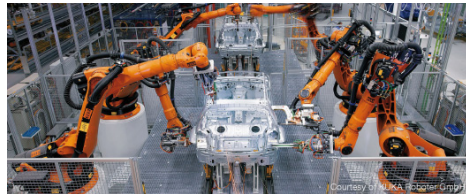
Why Study Programming?

- Robotics – wheeled, aerial, humanoid

Why Study Programming?

- Computer Vision – face recognition, image filtering

Why Study Programming?

- Movies – computer generated imagery, motion capture

Why Study Programming?

- Computer Games – computer graphics, audio programming, physics

Why Study Programming?

- Web Development – website design, information security
- Mobile Phones – short message service, mobile applications
- Big data Analytics – data processing, data analysis, machine learning
- Banking and Finance – financial systems simulation, policy modeling

Evolution of Programming Languages

- **Machine Language**
  - ▶ Lowest-level programming language
  - ▶ 0s and 1s
  - ▶ Easily understood by computers but is almost impossible for humans to use

- **Assembly Language**
  - ▶ English-like abbreviations such as MOV, ADD etc.
  - ▶ Translated into a machine language by a program called an assembler
  - ▶ Many instructions for even a simple task

- **High level Language**
  - ▶ Easier to understand for humans
  - ▶ A compiler is required to convert it into machine language
  - ▶ Single statement is enough to carry out many tasks

Why C++?

- **Why C++?**
  - Conciseness
  - Maintainabilty
  - Portability
- **Standardization**
  - ANSI/ISO standardization
  - Revisions – C++ 98, C++ 2003, C++ 2011, C++ 2014
- **C vs. C++**

- **What is Syntax?**
- **What is Algorithm?**
  Effective method defined in finite list of well-defined instructions for calculating a function.
- **What is Code?**
- **Process**

Criteria for Judging Code Quality

- Performance
- Simplicity (readability)
- Size
- Time taken