Outline

- Greeting
- Preface
- Lecture Goal
- Overview
- Q&A
Preface

- Excellent Choice
  - Computer Science & Engineering
    - High Demand
    - Interesting
    - Sense of Mission
Preface Cont’d
Lecture Goals

- Unix/Linux Centric Programming
- Considering the basic interaction in execution between the programs and the underlying computer systems
- Solving complicated problems by making C programs
- Understanding the Object-Oriented programming principles
- Acquiring good programming styles
- Learning C++ Object-Oriented programming methodologies
Why UNIX?

CLIENT PLATFORMS (CPU & OS)
- Windows 3.1, 95/98, NT, 2000 and XP
- DOS
- OS/2
- Solaris (Unix)
- IRIX (Unix)
- AIX (Unix)
- NEXTStep (Unix)
- Linux (Unix)

SERVER PLATFORMS (CPU & NOS)
- Windows NT and 2000
- Digital UNIX
- AIX (Unix)
- HP/UX (Unix)
- DEC VAX
- Compaq Alpha
- IBM RS/6000
- Silicon Graphics (MIPS)
- Digital Unix
- Tandem (MIPS)
- Encore (Alpha)
- DG/UX (Unix)
- Encore VAX
- DEC VAX
- Digital UNIX
- HP (PA-RISC)
- IBM RS/6000
- DEC VAX
- Encore VAX
- DEC VAX
- Digital UNIX
- Tandem (MIPS)
- Encore (Alpha)
- DG/UX (Unix)
- DEC VAX
- Digital UNIX
- Tandem (MIPS)
- Encore (Alpha)
- DG/UX (Unix)
Lecture Plan

- Program Execution on Unix/Linux
- Programming Tools
- Modularity and Abstraction in C
- C Pointers
- Memory Management in C
- File I/O in C, Libraries
- Object-Oriented Programming Principles
- Good Programming Style
Lecture Plan Cont’d

- C++ Basics, Data Types and Pointers
- Name Spaces, Operator Overloading
- Class Derivation and Inheritance
- C++ Templates, Standard Library
- Runtime Type Identification
- Exception Handling
- Design Patterns
- C vs C++, Review
Course Web Page URL

http://dcslab.snu.ac.kr/courses/cp2013s/