CS 241
Control Structures

Christopher A. Gantz

SPS Undergraduate Program
Regis University
cgantz@regis.edu
Lecture 1c

Overview of CS 241:
Problem Solving

Christopher A. Gantz
School of Professional Studies
Regis University
cgantz@regis.edu
Overview

• Reading
  – Nance textbook Pages 17-27

• Problem Solving
  – Program Development
  – Software Engineering
  – Software System Life Cycle
Program Development

• How do we solve a problem w/ SW?
  – Comprehensively understand the problem
  – Develop a solution/algorithm
  – Implement the solution/algorithm

• Why solving problems w/ SW is challenging!
  – Upfront planing is critical
  – Appropriate time management allocation to the process
  – Process execution requires a significant commitment to extreme precision and attention to detail
Program Development (Cont.)

- 6 ad hoc implementation best practices
  - Thorough and complete analysis of the problem
  - Design and development of an algorithm
  - Implementation of the algorithm as a SW program
  - Execution of the resulting SW program
  - Testing and verification of the functionality of the SW Program
  - Thorough internal and external documentation of the SW programs usage and functionality
Program Development (Cont.)

- What methodologies, tools and/or mechanisms exist to facilitate good program development?
  - Top Down design methodology
    - Process of continually subdividing the solution of the problem into subcomponents which represent a related collection of tasks and subtasks
    - Where each successive subdivision is referred to as a stepwise refinement
    - Process results in the identification of modules which are the tasks determined at stepwise refinement
Program Development (Cont.)

- Valuable artifacts of this methodology are:
  - Structure charts
    - Graphical representation that illustrates the relationship between modules
  - Identification of modules
  - Module specifications – formal descriptions of modules defining the following attributes
    - Data received (i.e. Input data)
    - Information returned (i.e. Output data)
    - Logic utilized to transform the input to output
Program Development (Cont.)

• Module specifications – formal descriptions of modules defining the following attributes (Cont.)
  – Logical assertions that define valid data states of the module
    • Pre-conditions
    • Post-conditions
    • Invariant
SW Engineering

- Process deployed to develop and maintain very large software systems
- Primarily used for software that is developed in teams
- Focuses on addressing large project
  - Management
  - Coordination
  - Design & Implementation
SW Engineering (Cont.)

- Benefits of deploying SW Engineering principles, methodologies, techniques and corresponding tools are:
  - Functionality
  - Maintainability
    - Reliability
    - Availability
    - Serviceability
  - Economic feasibility
SW Life Cycle

- A SW Life Cycle defines the following phases that all major SW systems experience
  - Analysis
  - Design
  - Coding/Implementation
  - Testing/Verification
  - Maintenance
  - Obsolescence [End of Life (EOL)]
Program Development Summary

• Successful SW development that implements a solution to a problem requires:
  – Significant and meticulous planning in order to understand the problem
  – Starting of the process as early as possible
  – Extreme precision and formality
  – Attention to detail with respect to design and implementation