ICSI333 System Fundamentals

Note: Students are expected to start the activities as soon as the description is available and seek feedback as needed. Although some activities are not graded for credit, they are contiguous study of the lecture or used as stepping-stones for the projects. Skipping any activities would impact the learning significantly.

Objectives:

- Discussion on previous labs and projects
- Share learning with others

Reading:

- Lecture notes
- Reading materials
- Labs and projects

Submission (5 points):

- No submission is required. The grade is determined by lab participation.

Instructions:

For this discussion, your co-instructor will facilitate a discussion on the previous labs and projects. Please bring your labs/projects to the lab. Please get ready to share your questions and learning with others.

The part I of the summary of the topics for this class has been shared. You may use it for study. The summary will be updated and shared. For your convenience, they are also listed below.

PART I:

- 00 - Introduction
  - Hardware and Software
    - Hardware and Software Interface
  - Computer Organization
- 01 – Number Systems
  - Number Systems
- 02 - Introduction to C Programming Language
  - C Programming Development Environment
  - Fundamentals of C
- 03 - Operators and Functions
  - Operators
    - Arithmetic Operators
    - Equality or Relational Operators
    - Increment and Decrement Operators
    - Logical Operators
    - Bitwise Operators
    - Assignment Operators
  - Decision Making, Loops
  - Functions
    - Function declarations, Function Definitions
    - Function Call by Value, Function Call by Reference
    - Scope Rules, Recursion

- 04 - Arrays Strings Structures and Unions
  - Arrays
  - Strings
  - Structures
  - Unions
- 05 - Dynamic Memory Allocation
  - Pointers
  - Dynamic Memory Allocation
  - Linked Lists
- 06 - Files and IO
  - Files
    - Create a sequential-access file for writing
    - Create a sequential-access file for reading
  - Input and Output
    - scanf and printf
    - getchar and putchar
    - getc and putc
    - fscanf and fprintf
    - Command-line arguments
    - I/O Streams
    - Redirection
- 07 - More about Strings and Pointer Arithmetic
  - Strings
    - String and character input/output functions: <stdio.h>
    - String manipulating functions: <string.h>
    - String conversion functions: <stdlib.h>
    - Character handling functions:<ctype.h>
  - Pointer Arithmetic
    - Pointer arithmetic operations:
      - Adding an integer to a pointer (+ or +=)
      - Subtracting an integer from a pointer (- or -=)
      - Incrementing (++) or decrementing (--)
      - Subtracting one pointer from another
    - Array and Pointer Interchangeability
- 08 - More about Memory Management
  - Dynamic Memory Management
    - Memory functions in string.h
      - memcpy, memmove, memcmo, memchr, memset
  - Multidimensional Matrix Memory Allocation
- 09 - Bit Fields
    - Defining Bit Fields
    - Unnamed Bit Fields
  - Bitwise Operators
    - Bitwise AND &, Inclusive OR |, Exclusive OR ^ and Complement Operators ~
    - Bitwise Left-Shift << and Bitwise Right-Shift >> Operators
    - Bitwise Assignment Operators
  - Big-Endian and Little-Endian
- 10 - The C Preprocessor
  - Preprocessor Directives
    - #include Preprocessor Directive
    - #define Preprocessor Directive: Symbolic Constants
    - #define Preprocessor Directive: Macros
  - Conditional Compilation

  - #error and #pragma Preprocessor Directives
  - # and ## Operators, Line Numbers, and Predefined Symbolic Constants
  - Assertions
- 11 - Compilation and Execution
  - Compilation
  - Execution

Linux Basic Commands

`ls [option(s)] [file(s)]`	If you run ls without any additional parameters, the program will list the contents of the current directory in short form. `-l` detailed list `-a` displays hidden files
`cp [option(s)] sourcefile targetfile`	Copies sourcefile to targetfile. `-I` waits for confirmation, if necessary, before an existing targetfile is overwritten `-r` copies recursively (includes subdirectories)
`mv [option(s)] sourcefile targetfile`	Copies sourcefile to targetfile then deletes the original sourcefile. `-b` creates a backup copy of the sourcefile before moving. `-I` waits for confirmation, if necessary, before an existing targetfile is overwritten.
`rm [option(s)] file(s)`	Removes the specified files from the file system. Directories are not removed by rm unless the option `-r` is used. `-r` deletes any existing subdirectories `-I` waits for confirmation before deleting each file
`cd [options(s)] [directory]`	Changes the current directory. cd without any parameters changes to the user's home directory.
`mkdir [option(s)] directoryname`	Creates a new directory.
`rmdir [option(s)] directoryname`	Deletes the specified directory, provided it is already empty.
`cat [option(s)] file(s)`	The cat command displays the contents of a file, printing the entire contents to the screen without interruption. `-n` numbers the output on the left margin
`cal`	Displays the calendar of the current month.
`date`	Displays current time and date.
`whoami`	Reveals the user who is currently logged in.
`whatis`	Gives a one-line description about the command. It can be used as a quick reference for any command.

`man`	Manual Pages, for more detailed information, Linux provides man pages and info pages. To see a command's manual page, man command is used.
`pwd`	Prints the absolute path to current working directory.
`vi`	A text editor for Linux operating system.
`gedit`	The default GUI text editor in the Ubuntu operating system.
`emacs`	Another text editor for Linux operating system.

Vi commands:

To save and quit:

Commands	Action
:wq	Save and quit
:w	Save
:q	Quit
:w fname	Save as fname
ZZ	Save and quit
:q!	Quit discarding changes made
:w!	Save (and write to non-writable file)

More commands:

Copy-pasting within the terminal: Ctrl+Shift+C/V

Resources and Credits:

- Teaching materials from Professor Kuperman at UAlbany
- Vi reference: https://www.ele.uri.edu/faculty/vetter/Other-stuff/vi/vi-quick-ref.pdf