CIS22a: Beginning Programming methodologies in C++

Recommended Text: Starting out with C++, Tony Gaddis, Any Edition, ISBN-13 978-0-13-257625-3

Assignment Due dates : Apr, 18,30, May 9,21,30, Jun 11,20

Midterm ONE: May 2, 2018

Midterm TWO: May 30, 2018

Finals: Jun 25, 2018

This course is designed to introduce computer programming in C++. Emphasis will be placed on structured procedural programming with an introduction to object-oriented programming. C++ data types, identifiers, programming constructs, functions, pointers, structures, Classes and arrays will be covered in detail with examples.

Course dates and Schedule: http://deanza.fhda.edu/calendar/spring/dates.html

Contact Information

- Instructor: L Krishnamurthy
- Contact Email: krishnamurthylalitha at fhda dot edu
- Instructor Online hours: Thurs : 8PM -9.15PM
- Instructor Office Hours: Tuesday : 730pm - 830PM

Students, please read the following and plan accordingly

- All communication with instructor is via Canvas for assignments and quizzes. Questions, comments will be addressed face to face in class, during class hours.
- **Your mail subject and body must be in English - No other languages will be accepted. Email will be discarded and you will not get any points !! Please note this to ensure success.**
- Class starts at 8.10pm. Please wait till then for your questions/comments
- Recording of audio/video of my lectures using your smart-device is NOT permitted.
- Your internet activities are monitored in classroom computers. Please note this

- Students who constantly chit-chat during lectures will be asked to leave the classroom.
- There is no "make-up" for FINALS or MIDTERMs -ie if you miss it you lose the points. NOTE THIS.
- Do keep all your communication about graded assignments, midterm, etc till end of class. Do not discard or lose them (I might use them to verify grades at end of quarter)

Rules of assignment submission below

- Assignment can be submitted once via Canvas.
- Please use your full name (not nickname or pen-names) as it appears in class roster for any (email) communication with instructor, Also use your first and last name correctly (refer to class roster as to how it is specified).
- Use Canvas for assignment submission.

SLO for this class

- Student Learning Outcome (1): Design solutions for introductory level problems using appropriate design methodology incorporating elementary programming constructs.
- Student Learning Outcome (2): Create algorithms, code, document, debug, and test introductory level C++ programs.
- Student Learning Outcome (3): Read, analyze and explain introductory level C++ programs.

Objectives the lectures will help accomplish

- Student will learn the difference between procedural and object oriented programming.
- Student will learn the software life-cycle steps including design, development, styles, documentation, testing, and maintenance in the creation of program.
- Student will learn to use the C++ environment in the development and testing of programs.
- Student will learn declaring identifiers of different data types, Use data types to declare variables in C++ programs.
- Student will learn to apply input and output functions to read data using keyboard and output to screen.
- Student will learn to use expressions, statements and operators to construct program building blocks that compute values.
- Student will learn to program control structures to break up flow of program execution and conditionally execute blocks of code.

- Student will learn to implement functions/methods in programs for clarity and efficiency in code development.
- Student will learn to develop programs using functions that enable input and output with text files.
- Student will learn to program usage of arrays to process variety of data problems.

Week 1: Read Chapter 1 and 2	Programming language overview, Compare and contrast procedural versus object oriented programming languages. describe flow chart, Software development cycle, Testing and execution. Describe a general C++ program structure and its syntax.
Week 2: Read Chapter 1 and 2	Why C++, What is C++, How C++, Structure of C++ program, C++ data types, variables, printing to console/standard output, reading from console/standard input (cin/cout). Describe C++ include directive, Cout/Cin and its syntax.
Week 3: Read Chapter 3 and 4	C++ expressions, Formatting input/output, C++ operators, relational operators, control statements, if, switch, examples. Illustrate constants, variables, definition and declaration, Discuss primitive data types. Discuss operators, expression formats.
Week 4: Read Chapter 3 and 4	C++ operators review, relational operators, control statements, if, switch, examples. Decision statemnets and their purpose and structure. Syntax of if switch control statements and their formats, Relational operators Logical operators, Expanding the 'if' with 'else' and 'else if' , The 'switch' statement
Week 5: Read Chapter 5 and 6	C++ loops, while, do-while, for loop, C++ Functions
Week 6: Read Chapter 5 and 6	Review Chapters 1 - 5. Discuss various programs to help seal understanding of C++ syntax. Illustrate using programs from textbook Chapters 1-5 as a review
Week 7:	Introduce C++ blocks and functions. Midterm:
Week 8: Read Chapter 5 and 6	C++ functions/methods in programs for clarity and efficiency in code development. Discuss Passing arguments to parameters, Pass by value , Pass by reference , Software engineering: modular programming to help understand the purpose of functions
Week 9: Read Chapter 7	Introduction to One-dimensional Arrays, Accessing array elements - Initialization
Week 10: Read Chapter 7	Accessing array elements - Initialization , Processing array contents Application of arrays - Sequential search, Selection sortC++ pointers, Pointers and Arrays
Week 11: Read Chapter 10	Review of arrays and functions; String API, C-Strings examples, Introduce C++ pointers
Week 12	Review and Finals

Grading

a. Assignments worth 70 points, Two midterms, 10 points each, Finals 10 points.