

1. Course number and name

CptS 121: Program Design and Development C/C++

2. Credits and contact hours

4 credits, 3 lecture hours, 3 lab hours

3. Instructor's or course coordinator's name

Andy O'Fallon

4. Textbook, title, author, and year

J.R. Hanly and E.B. Koffman. 2016. *Problem Solving & Program Design in C* (8th ed.). Pearson Education, Inc., Addison-Wesley.

Other supplemental materials

P.J. Deitel and H.M. Deitel, 2016. *C: How to Program* (8th ed.). Pearson Education, Inc. Prentice Hall.

5. Specific course information

- a. *Catalog description:* Formulation of problems and top-down design of programs in a modern structured language (C/C++) for their solution on a digital computer.
- b. *Prerequisites or co-requisites:* MATH 108, 171, 172, 182, 201, 202, 206, or 220, each with a C or better, or a minimum ALEKS math placement score of 80%, or adequate CPT S placement test score determined by the department.

6. Specific goals for the course

By the end of the course, students will be able to

- Design, implement, and test a program by applying modern tools and techniques (1a, 1b, 1d, 1e, 6a, 6b, 7b, 7d, 7f, 7g).
- Analyze a specification of a problem of modern complexity and construct a structured, elegant C program that solves the problem (1a, 1b, 1d, 1e, 6a, 6b, 7b, 7d, 7f, 7g).
- Document C programs applying the guidelines specified in a standards and styles guide (3a)
- Design and articulate solutions to lab problems with classmates (1a, 1b, 1d, 1e, 3a, 3b, 5b, 5g).
- Identify and implement test cases to edge scenarios in pseudocode and/or C code (6b, 6d).
- Identify, analyze, and solve C code interview questions in prep for internships (1a, 1b, 1c, 1d, 1e).

7. Brief list of topics to be covered

- Introduction to Algorithms
- Software Development Process
- Variables, Data Types, and Operators
- Numeric Expressions

- Functions (standard library and user-defined)
- File Processing
- Selection Structures
- Loops (loop patterns)
- Modular Programming
- Arrays
- Pointers
- Strings
- Structures
- Recursion
- Bit Manipulation
- Dynamic Data Structures
- Command Line Arguments
- Testing and Debugging
- Macros
- Multi-file Programs
- Problem Solving with Algorithms
- Memory Organization
- Pseudocode
- Software Design and Engineering Concepts
- Problem Solving Strategies