

# CS G102: COMPUTER SOFTWARE DEVELOPMENT, INTRODUCTION

---

Item	Value
Curriculum Committee Approval Date	12/03/2024
Top Code	070700 - Computer Software Development
Units	3 Total Units
Hours	90 Total Hours (Lecture Hours 36; Lab Hours 54)
Total Outside of Class Hours	0
Course Credit Status	Credit: Degree Applicable (D)
Material Fee	No
Basic Skills	Not Basic Skills (N)
Repeatable	No
Open Entry/Open Exit	No
Grading Policy	Standard Letter (S), • Pass/No Pass (B)

## Course Description

This course will introduce students to the basic principles of computers and software development. Computer architecture concepts and principles of operation, operating systems basics, file management, and software development principles will be discussed, demonstrated and practiced. Common programming languages, including Java, C, C++, and Python will be discussed and small programs in each language will be written. ADVISORY: Course taught at the level of intermediate algebra or appropriate math placement. Transfer Credit: CSU.

## Course Level Student Learning Outcome(s)

1. Course Outcomes
2. Utilize the basic functional building blocks of operating systems.
3. Create programs using different languages.
4. Describe the fundamentals of a computer system.

## Course Objectives

- 1. Summarize the fundamentals of computer architecture.
- 2. Describe the basic building blocks of operating systems.
- 3. Utilize file management techniques.
- 4. Categorize common software development methodology.
- 5. Describe the differences between the common programming languages.
- 6. Utilize software development tools.
- 7. Create simple algorithms and applications
- 8. Describe the fundamentals of computer networking.
- 9. Utilize the Internet and the World Wide Web.

## Lecture Content

Basics of computer architecture and operation RAM (Random Access Memory), ROM (Read Only Memory), CPU (Central Processing Unit), storage units, and peripherals Memory models and related performance characteristics Application execution and data flow Functional components of computer operating systems File management techniques and associated software Introductory concepts of computer networking Internet, World Wide Web, and various tools and services Software development tools and methodologies Overview of common programming languages Source code comparison Compilers and compilation tools Optimizers Linkers Run time engines Libraries Modularization Software components Services Computer application development concepts Program flow and data representation Branching Looping Arrays Strings Pointers Computer ethics, cybersecurity, privacy, and other social implications Artificial Intelligence (AI) impact on software development

## Lab Content

Software Development Life Cycle (SDLC) Waterfall, spiral, incremental models Agile development Development tools: Integrated Development Environments (IDEs) Open source vs. proprietary Programming concepts implementation Branching Looping Arrays Strings Pointers Computer applications Problem analysis Solution design Programming code Testing and integration Software deployment and support Internet and web development concepts Browsers Front-end tools Back-end components

## Method(s) of Instruction

- Lecture (02)
- DE Live Online Lecture (02S)
- DE Online Lecture (02X)
- Lab (04)
- DE Live Online Lab (04S)
- DE Online Lab (04X)

## Reading Assignments

Textbook and Websites

## Writing Assignments

Students will be required to complete computer operation and introductory computer programming projects presented to them in the form of business automation problems requiring solution implementation. Students will be required to write documentation on their projects.

## Out-of-class Assignments

An optional library research paper will promote further study and research in current Component Programming or other related topics selected by the student and approved by the instructor.

## Demonstration of Critical Thinking

Students will analyze requirements and select the appropriate programming tools for an efficient solution implementation. Testing and debugging will require students to perform data tracing and error isolation during program execution.

## Required Writing, Problem Solving, Skills Demonstration

Students will be required to complete computer operation and introductory computer programming projects presented to them in the form of business automation problems requiring solution

implementation. Students will be required to write documentation on their projects.

### **Eligible Disciplines**

Computer science: Master's degree in computer science or computer engineering OR bachelor's degree in either of the above AND master's degree in mathematics, cybernetics, business administration, accounting or engineering OR bachelor's degree in engineering AND master's degree in cybernetics, engineering mathematics, or business administration OR bachelor's degree in mathematics AND master's degree in cybernetics, engineering mathematics, or business administration OR bachelor's degree in any of the above AND a master's degree in information science, computer information systems, or information systems OR the equivalent. Note: Courses in the use of computer programs for application to a particular discipline may be classified, for the minimum qualification purposes, under the discipline of the application. Master's degree required.

### **Textbooks Resources**

1. Required Schneider, G.M., Gersting, J.L.. Invitation to Computer Science (Classic), 8th ed. Cengage, 2019 Rationale: Textbook content matches the course lecture and lab content.