

Course Number and Title: CSC 210 Systems Programming

Campus Location:

Georgetown, Dover, Wilmington

Effective Date:

2021-51

Prerequisite:

CSC 164

Co-Requisites:

None

Course Credits and Hours:

3.00 credits

2.00 lecture hours/week

2.00 lab hours/week

Course Description:

This course introduces principles of computer systems programming software and hardware platforms to achieve efficient resource usage. The focus is on low level programming of system hardware. Topics include traditional C programming language, memory management, awareness of system constraints, and interfacing. Students work in a Linux environment, apply concepts in software development such as pointers and memory management, and complete a programming embedded systems project.

Required Text(s):

Obtain current textbook information by viewing the [campus bookstore - https://www.dtcc.edu/bookstores](https://www.dtcc.edu/bookstores) online or visit a campus bookstore. Check your course schedule for the course number and section.

Additional Materials:

None

Schedule Type:

Classroom Course

Video Conferencing

Web Conferencing

Hybrid Course

Disclaimer:

None

Core Course Performance Objectives (CCPOs):

1. Implement a system programming language such as standard C to solve problems. (CCC 2, 6; PGC 1)
2. Create and modify Linux operating system scripts to interact with Linux devices. (CCC 1, 2, 6; PGC 3, 5)
3. Apply system programming language to interface hardware in a Linux environment. (CCC 1, 2, 6; PGC 3, 5)

See Core Curriculum Competencies and Program Graduate Competencies at the end of the syllabus. CCPOs are linked to every competency they develop.

Measurable Performance Objectives (MPOs):

Upon completion of this course, the student will:

1. Implement a system programming language such as standard C to solve problems.
 1. Develop programs using variables and control structures (sequence, branch, and loop)
 2. Construct programs by modulating components using functions and structures.
 3. Create programs in C language with pointers to write efficient programs
2. Create and modify Linux operating system scripts to interact with Linux devices.
 1. Perform operations in the Linux operating system with commands.
 2. Connect devices using Linux commands.
 3. Write and debug Linux scripts to establish, maintain, and troubleshoot device interfaces.
3. Apply system programming language to interface hardware in a Linux environment.
 1. Create programs to manage Linux file systems, including accessing and using device files.
 2. Apply memory management concepts in system programming in Linux platform.
 3. Compare and contrast different common hardware interface types.
 4. Write and debug programs in a Linux environment using different interface types
 5. Write and debug programs to connect a hardware device.

Evaluation Criteria/Policies:

The grade will be determined using the Delaware Tech grading system:

90	-	100	=	A
80	-	89	=	B
70	-	79	=	C
0	-	69	=	F

Students should refer to the [Student Handbook - https://www.dtcc.edu/handbook](https://www.dtcc.edu/handbook) for information on the Academic Standing Policy, the Academic Integrity Policy, Student Rights and Responsibilities, and other policies relevant to their academic progress.

Final Course Grade:

Calculated using the following weighted average

Measure	Percentage
3 Exams w/Lab & Final Exam equally weighted. (summative)	40%
2 – 4 Labs equally weighted. (formative)	20%
2 – 4 Programming Assignments equally weighted. (formative)	20%
Final Project (summative)	20%

Core Curriculum Competencies (CCCs are the competencies every graduate will develop):

1. Apply clear and effective communication skills.
2. Use critical thinking to solve problems.
3. Collaborate to achieve a common goal.
4. Demonstrate professional and ethical conduct.
5. Use information literacy for effective vocational and/or academic research.
6. Apply quantitative reasoning and/or scientific inquiry to solve practical problems.

Program Graduate Competencies (PGCs are the competencies every graduate will develop specific to his or her major):

1	Design and implement moderate to complex programs that meet specifications, perform reliably, and are maintainable using the principles of software engineering.
2	Apply object-oriented design principles to software analysis and programming.
3	Analyze currently available operating systems and software development platforms to design and implement software applications that are effective and secure.
4	Develop programs in assembly language that directly address the computer architecture.
5	Develop technical documentation to meet end user requirements.

Disabilities Support Statement:

The College is committed to providing reasonable accommodations for students with disabilities. Students are encouraged to schedule an appointment with the campus Disabilities Support Counselor to request an accommodation needed due to a disability. A listing of campus Disabilities Support Counselors and contact information can be found at the [disabilities services - https://www.dtcc.edu/disabilitysupport](https://www.dtcc.edu/disabilitysupport) web page or visit the campus Advising Center.