



Course Number and Title: CSC 214 Computer Science III

Campus Location:

Georgetown, Dover, Wilmington

Effective Date:

2020-51

Prerequisite:

CSC 164

Co-Requisites:

none

Course Credits and Hours:

4.00 credits

3.00 lecture hours/week

2.00 lab hours/week

Course Description:

This course, the third in a series, provides a foundation in computer science. Students develop intermediate-to-advanced programming skills using a language that supports an object-oriented approach. Emphasis is placed on data structures, algorithmic analysis, software engineering principles, software and information assurance, and professionalism.

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:

USB flash-drive for lab work

Schedule Type:

Classroom Course

Disclaimer:

None

Core Course Performance Objectives (CCPOs):

1. Assess and examine major software failures in history. (CCC 1, 5; PGC 3)
2. Differentiate among the principal types of software engineering methodologies. (CCC 1,5, 6; PGC 1, 2, 3, 5)
3. Develop moderately complex programs. (CCC 2, 6; PGC 1, 2, 5)
4. Create and apply algorithms and data structures in a business application problem. (CCC 2, 6; PGC 1, 2, 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. Assess and examine major software failures in computer history.
 1. List disasters caused by software failures in the history of computing field.
 2. Assess the impact of the disasters.
 3. Identify the causes of the failures in terms of software engineering principles.
2. Differentiate among the principal types of software engineering methodologies.
 1. Analyze and identify system elements and their characteristics.
 2. Identify and examine the issues related to software development life cycle.
 3. Identify the software life cycle model, and specify the timeline for each phase.
3. Develop moderately complex programs.
 1. Produce user interfaces that incorporate modularity and error handling events.
 2. Use event-driven programming techniques, including graphics application programming interface (API), event-handling methods, and exception handling in developing applications according to given specifications.
 3. Create programs implementing searching and sorting algorithms, and evaluate efficiency of the implementation.
4. Create and apply algorithms and data structures in a business application problem.
 1. Create program specifications for a business application problem.
 2. Create object-oriented programs that implement the program specifications.
 3. Compare and contrast different implementations of the programs.
 4. Develop applications of medium complexity employing software reuse, design patterns, parametric polymorphism (templates or generics), code libraries, container classes, and iterators.

Evaluation Criteria/Policies:

Students must demonstrate proficiency on all CCPOs at a minimal 75 percent level to successfully complete the course. The grade will be determined using the Delaware Tech grading system:

92	-	100	=	A
83	-	91	=	B
75	-	82	=	C
0	-	74	=	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

Evaluation Measure	Percentage of final grade
Lecture Exams: 3 exams weighted at 10% each (summative)	30%
Final Exam (summative)	10%
Research paper (summative)	10%
Programming Assignments: 5 weighted at 7% each (summative)	35%
Assessments: quizzes, labs, and homework assignments (formative) (equally weighted)	15%
TOTAL	100%

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.