



## Course Number and Title: CSC 114 Computer Science I

**Campus Location:**

Georgetown, Dover, Wilmington

**Effective Date:**

2018-51

**Prerequisite:**

ENG 090 or ENG 091, MAT 020, SSC 100 or concurrent

**Co-Requisites:**

None

**Course Credits and Hours:**

4.00 credits

3.00 lecture hours/week

2.00 lab hours/week

**Course Description:**

This course introduces the fundamental concepts of programming. Topics include data types, control structures, functions, arrays, files, and the mechanics of running, testing, debugging, and documenting programs. Additionally, the concepts of data abstraction and recursion are introduced. Students employ fundamental concepts to create and assess simple programs.

**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

**Disclaimer:**

None

**Core Course Performance Objectives (CCPOs):**

1. Practice professional behaviors in response to ethical issues inherent in computing. (CCC 3, 4; PGC 1, 3)
2. Employ algorithms for solving simple problems, and trace the execution of computer programs. (CCC 2, 6; PGC 1, 3)
3. Compare and contrast the primitive data types of a programming language, describe how each is stored in memory, and identify the criteria for selection. (CCC 2, 6; PGC 1, 3)
4. Analyze and apply the program development process to problems that are solved using fundamental programming constructs. (CCC 2, 6; PGC 1, 3)
5. Decompose programming problems, and produce solutions using a series of linked modules. (CCC 2, 6; PGC 1, 3)
6. Describe the language translation phases of edit, compile/interpret, link, and execute. (CCC 1; PGC 1, 3)
7. Employ fundamental programming concepts in developing programs. (CCC 2, 6; PGC 1, 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. Practice professional behaviors in response to ethical issues inherent in computing.
  1. Identify ethical issues inherent in a programming assignment.
  2. Locate and apply professional standards bearing on a particular ethical issue.
  3. Write programs that are consistent with professional standards of conduct.
2. Employ algorithms for solving simple problems, and trace the execution of computer programs.
  1. Explain and use algorithms to solve common computing tasks such as sorting.
  2. Create simple programs implementing common algorithms.
  3. Locate and employ libraries of algorithms.
3. Compare and contrast the primitive data types of a programming language, describe how each is stored in memory, and identify the criteria for selection.
  1. Discuss the representation and use of primitive data types and built-in data structures.
  2. Select primitive data types appropriate to the solution of a programming problem.
4. Analyze and apply the program development process to problems that are solved using fundamental programming constructs.
  1. Analyze a problem, and describe the data required to solve the problem.
  2. Prepare a brief description of the problem.
  3. Propose a data model.
  4. Illustrate the data definitions and problem description with examples.
  5. Create a set of instructions based on the analysis of a problem.
  6. Implement the instructions based on the analysis of a problem in code.
  7. Employ test data to test code.
  8. List the recommended steps in problem solving.
  9. Organize work in accordance with the program development process.
5. Decompose programming problems, and produce solutions using a series of linked modules.
  1. Modify and expand short programs using control structures and functions.
  2. Explain the divide-and-conquer approach.
  3. Describe and explain parameter passing.
  4. Employ parameter passing in simple programs.
6. Describe the language translation phases of edit, compile/interpret, link, and execute.
  1. Discuss the error conditions that may occur at each phase.
  2. Discuss techniques for handling error conditions occurring at each phase.
7. Employ fundamental programming concepts in developing programs.
  1. Use simple object-oriented programming concepts.
  2. Use recursion to solve simple programming tasks.
  3. Use techniques such as iteration, parameter passing, and propositional and predicate logic to solve simple programming assignments.
  4. Employ data and procedural abstraction in solving simple programming assignments.

### 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.

### 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.