



**ESSEX COUNTY COLLEGE**  
**Engineering and Computer Science Division**  
**CSC 121 – Computer Science I**  
**Course Outline**

**Course Number & Name:** CSC 121 Computer Science I

**Credit Hours:** 4.0    **Contact Hours:** 4.0    **Lecture:** 4.0    **Lab:** N/A    **Other:** N/A

**Prerequisites:** Grades of "C" or better in MTH 100 and CSC 100 or placement

**Co-requisites:** None

**Instructor:** TBA

**Email:** TBA

**Office Hours:** By appointment

---

**Course Description:** This course serves as an introduction to the concepts and methodologies fundamental to computer science. Emphasis is placed upon object-oriented design and analysis with a thorough discussion of the concepts and principles associated with object-oriented programming. A high level object-oriented language is utilized for programming assignments and to illustrate conceptual material. It is recommended that a student be enrolled concurrently in either MTH 113 or MTH 119 to derive the most benefit from the course.

**Textbook:** *C++ Programming: Program Design Including Data Structures* (5th edition) by D S Malik: Thompson Course Technology, 2004.

Additional required course materials: USB flash memory and a note book for taking notes.

**General Education Goals:** Upon successful completion of this course, students should be able to do the following:

1. demonstrate knowledge of basic concepts and methodologies of computer science;
2. define algorithm;
3. utilize various problem-solving techniques to create algorithms
4. design application programs to implement algorithms in an object-oriented language;
5. use a computer system as a tool for problem solving; and

6. communicate accurate computing terminology and notation in written and/or oral form.

**Measurable Course Performance Objectives (MPOs):** Upon successful completion of this course, students should specifically be able to do the following:

1. Demonstrate knowledge of basic concepts and methodologies of computer science:
  - 1.1 *define and discuss relevant 'computer science' words and concepts;*
  - 1.2 *explain the relation of the 'computer science' concepts to computing; and*
  - 1.3 *use computer science methodologies to solve problems*
  
2. Define algorithm:
  - 2.1 *write a definition of algorithm;*
  - 2.2 *identify examples of algorithms and explain how they conform to the definition of algorithm; and*
  - 2.3 *analyze solutions to problems that are not algorithms and explain why they are not*
  
3. Utilize various problem-solving techniques to create algorithms:
  - 3.1 *create and use flow charts;*
  - 3.2 *determine and use procedural methods; and*
  - 3.3 *outline and use object-oriented methods*
  
4. Design application programs to implement algorithms in an object-oriented language:
  - 4.1 *discuss and apply the basic structures of programming language; and*
  - 4.2 *define 'class' and use 'classes' to implement an ADT*
  
5. Use a computer system as a tool for problem solving:
  - 5.1 *using an integrated programming environment, write and complete a program; and*
  - 5.2 *execute and test the solution to the problem*
  
6. Communicate accurate computing terminology and notation in written and/or oral form:
  - 6.1 *document the programs so they are self explanatory;*
  - 6.2 *use correct computing terminology on written and oral assignments/assessments*

**Methods of Instruction:** Instruction will consist of lectures, laboratory demonstrations and assignments, and programming examples.

**Outcomes Assessment:** Exam questions cover the major topics of the course and show the extent of student mastery of the course material. Checklist rubrics are used to evaluate the projects for the presence of course objectives. Data is collected and analyzed to determine the level of student performance on these assessment instruments in regards to meeting course objectives. The results of test performance and project completion are used to decide if there is sufficient mastery to move on to the next section.

**Course Requirements:** All students are required to:

1. Maintain regular attendance and take part in class discussions.
2. Complete assigned homework and projects on time.
3. Take all exams as scheduled.

**Attendance Policy:** Regular and prompt attendance is essential for academic success. Faculty members take attendance at each scheduled class session. Students are expected to attend and be on time for all classes. Individual faculty members may establish specific attendance policies. Attendance records will be turned in to the appropriate Division/Department Chair and/or Program Director at the end of the term and in the interim upon request. Any students with more than three unexcused absences will automatically fail the course.

**Methods of Evaluation:** Final course grades will be computed as follows:

<b>Grading Components</b>	<b>% of final course grade</b>
<ul style="list-style-type: none"><li>• <b>Homework, class participation and attendance</b> Students must practice skills on their own by doing homework to be able to master course objectives. Homework assignments relate to these objectives. Attendance and class participation are necessary for students to benefit from the guidance of the instructor.</li></ul>	<b>10 %</b>
<ul style="list-style-type: none"><li>• <b>Projects</b> (due dates specified by the instructor) Projects will show evidence of the extent to which students meet course objectives. Students should show that they have synthesized a combination of concepts.</li></ul>	<b>35%</b>
<ul style="list-style-type: none"><li>• <b>Tests</b> (dates specified by the instructor) Tests will provide evidence of the extent students have mastered course objectives.</li></ul>	<b>25%</b>
<ul style="list-style-type: none"><li>• <b>Final Exam</b> Tests will provide evidence of the extent students have mastered course objectives.</li></ul>	<b>30%</b>

**NOTE:** The instructor will provide specific weights, which lie in the above-given ranges, for each of the grading components at the beginning of the semester.

**Grading System:**

<b>A</b>	<b>90% - 100%</b>	<b>Superior</b>
<b>B+</b>	<b>87% - 89%</b>	<b>Very Good</b>
<b>B</b>	<b>80% - 86%</b>	<b>Good</b>
<b>C+</b>	<b>77% - 79%</b>	<b>Above Average</b>
<b>C</b>	<b>70% - 76%</b>	<b>Satisfactory</b>
<b>D</b>	<b>60% - 69%</b>	<b>Passing</b>
<b>F</b>	<b>59% - 0</b>	<b>Failing</b>

**Academic Integrity:** Dishonesty disrupts the search for truth that is inherent in the learning process and so devalues the purpose and the mission of the College. Academic dishonesty includes, but is not limited to, the following:

- plagiarism – the failure to acknowledge another writer’s words or ideas or to give proper credit to sources of information;
- cheating – knowingly obtaining or giving unauthorized information on any test/exam or any other academic assignment;
- interference – any interruption of the academic process that prevents others from the proper engagement in learning or teaching; and
- fraud – any act or instance of willful deceit or trickery.

Violations of academic integrity will be dealt with by imposing appropriate sanctions. Sanctions for acts of academic dishonesty could include the resubmission of an assignment, failure of the test/exam, failure in the course, probation, suspension from the College, and even expulsion from the College.

**Student Code of Conduct:** All students are expected to conduct themselves as responsible and considerate adults who respect the rights of others. Disruptive behavior will not be tolerated. All students are also expected to attend and be on time all class meetings. No cell phones or similar electronic devices are permitted in class. Please refer to the Essex County College student handbook, *Lifeline*, for more specific information about the College’s Code of Conduct and attendance requirements.

**Course Content Outline:** This is a tentative course schedule, the instructor reserve the right to make changes on it to make it better for the student`s development. Notice will be given should any changes take place.

### **Class Meeting**

<b>(120 minutes)</b>	<b>Topics to be Covered/ Class Activitie</b>
1 – 2	Orientation&Course Introduction Review of Class Syllabus 0.1.1 What is Computer Science 0.1.2 History of Computing 0.1.3 Introduction to Computer Systems 0.1.4 Algorithms
3 – 4	Basic Data Representation Basic Elements of C++ (Chapter 2) Types and Expressions Types and Declarations Data Representation
5 – 6	Operations and Expressions Numeric Expressions Boolean Expressions Character Expressions Assignment Expressions
7 – 8	Computers Communicate with World Input/Output (Chapter 3) I/O Expressions
9 – 10	Control of the Flow and Issues of Computability Selection (Chapter 4) If Statement Switch Statement Conditional Expressions

**Class Meeting****(120 minutes)****Topics to be Covered/ Class Activities**

---

11 – 13	Computability and Repetition Repetition (Chapter 5) The for Loop The while Loop The do Loop Input Loops
14	Test 1
15 – 17	Using Functions to Solve Problems Functions (Chapter 6) Function Definition Function Prototypes Local Variables Functions that Return Nothing Functions that Use Selection Functions that Use Repetition
18 – 19	More Functions Functions in Depth (Chapter 7) Parameters in Depth Inline Functions Recursion
20	Test 2
21 – 23	Data types and ADT Arrays (Chapter 9) C-Style Arrays Sorting Searching

**Class Meeting****(120 minutes)      Topics to be Covered/ Class Activitie**

---

24 – 25	More Dimensions Multidimensional Arrays (Chapter 13) C-Style Multidimensional Arrays
26	Final Examination

**CSC 121 Suggested Projects**

NOTE: Suggested projects listed below are illustrative of assignments that are distributed through the course to correspond to the topics being discussed in class.

**Project 1:** “Hello world”

**Project 2:** Mathematical operations

**Project 3:** Reading files

**Project 4:** Using nested “if” statements and switch

**Project 5:** Using loops (three kinds)

**Project 6:** Functions – square root, predefined and user-defined

**Project 7:** Paper on free software movement

**Project 8:** Sorting and searching and efficiency

**Project 9:** Use of recursion in solving problems

**Project 10:** Comparison of recursion and iteration