I. INTRODUCTION

A. This course focuses on the object-oriented programming paradigm, emphasizing the definition and use of classes along with fundamentals of object-oriented design. The course includes basic analysis of algorithms, searching and sorting techniques, and an introduction to software engineering processes. Students will apply techniques for testing and debugging software.

B. This course serves as a required or elective course on various degree plans. Curriculum plans for degrees and certificates are listed in the current Central Texas College catalog.

C. This course, in conjunction with the other courses listed on the degree plan, serves as preparation for careers in computer programming.

D. This course may be delivered as a traditional lecture/lab, non-self-paced online, or blended course.

E. Prerequisites: COSC 1336 Programming Fundamentals I.

II. LEARNING OUTCOMES

Upon successful completion of this course, Programming Fundamentals II, the student will be able to:

A. Demonstrate a basic understanding of object-oriented programming by using structs and classes in software projects. C1, C7, C8, C16, C18, F1, F3, F8)

B. Use object-oriented programming techniques to develop executable programs that include elements such as inheritance and polymorphism. (C5, C6, C8, F2, F11)

C. Document and format code in a consistent manner. (C1, C8, C16, C18, F1, F3, F8)

D. Apply basic searching and sorting algorithms in software design. (C1, C7, C8, C16, C18, F1, F3, F8)
E. Apply single- and multi-dimensional arrays in software. (C5, C6, C8, F2, F11)
F. Use a symbolic debugger to find and fix runtime and logical errors in software. (C1, C8, C16, C18, F1, F3, F8)
G. Demonstrate a basic understanding of programming methodologies, including object-oriented, structured, and procedural programming. (C1, C7, C8, C16, C18, F1, F3, F8)
H. Describe the phases of program translation from source code to executable code. (C5, C6, C8, F2, F11)

III. INSTRUCTIONAL MATERIALS

A. The instructional materials identified for this course are viewable through www.cted.edu/books

B. Lecture Classes also require at least one USB storage device. Online students may use cloud based storage.

IV. COURSE REQUIREMENTS

A. Attend both lecture and lab or in the case of online delivery, be actively engaged in Blackboard and maintain constant progress.

B. Be prepared to participate in discussion, team projects/assignments and take unannounced assessments relating to the lecture materials.

C. Complete all exams/assessments.

D. Submit all assignments on time.

V. ASSESSMENTS

A. Student content mastery will be evaluated in the following areas:
   • Assessments (midterm exam, quizzes, projects, etc.)
   • Final Assessment (final exam and/or semester project, participation)

B. Scheduled and unscheduled quizzes will be given at the discretion of the instructor.

C. Exams/assessments may be composed of both subjective and objective questions plus computer output.

D. A student must take all exams/assessments. No make-up exams/assessments will be given. Both online and on campus students who know in advance that they will
be absent due to school sponsored trips, military duty or orders, or any other valid reason, must arrange to take an early exam/assessment. Unexpected absences due to illness or other extenuating circumstances will require the student to see the instructor about make-up work in lieu of the missed exam/assessment.

E. Students with unexcused absences will be given a zero for any missed work.

VI. SEMESTER GRADE COMPUTATIONS

<table>
<thead>
<tr>
<th>Course Requirements</th>
<th>Points</th>
<th>Points</th>
<th>Grade</th>
<th>Quality Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignments</td>
<td>300</td>
<td>900</td>
<td>700 - 799</td>
<td>600 - 699</td>
</tr>
<tr>
<td>Assessments</td>
<td>300</td>
<td>800</td>
<td>800 - 899</td>
<td>400</td>
</tr>
<tr>
<td>Final Assessment</td>
<td>400</td>
<td>400</td>
<td>600 - 699</td>
<td>400</td>
</tr>
<tr>
<td>TOTAL</td>
<td>1000</td>
<td>1000</td>
<td>900 - 1000</td>
<td>900</td>
</tr>
</tbody>
</table>

VII. NOTES AND ADDITIONAL INSTRUCTIONS FROM THE INSTRUCTOR

A. Information on the following Academic Policies, as described in the CTC Course Catalog will be followed:
   1. Withdrawals
   2. Grading
   3. Class Attendance and Course Progress
   4. Scholastic Honesty

B. Cell Phones and Pagers: Students will silence cell phones and mobile devices while in the classroom or lab.

C. Americans with Disabilities Act (ADA): Disability Support Services provide services to students who have appropriate documentation of a disability. Students requiring accommodations for class are responsible for contacting the Office of Disability Support Services (DSS) located on the central campus. This service is available to all students, regardless of location. Review the website at www.ctcd.edu/disability-support for further information. Reasonable accommodations will be given in accordance with the federal and state laws through the DSS office.

D. Instructor Discretion: The instructor reserves the right of final decision in course requirements and may make changes to the course outline and/or assignments as needed.
E. **Civility**: Individuals are expected to be aware of what a constructive educational experience is and be respectful of those participating in a learning environment. Failure to do so can result in disciplinary action up to and including expulsion.
VIII. COURSE OUTLINE

A. Unit One: Introduce course requirements and objectives. Introduce students to the laboratory and use of the microcomputer.

1. Learning Outcomes: Upon successful completion of this unit the student will be able to:
   a. Understand course requirements as defined in the syllabus and reviewed by the instructor
   b. Demonstrate ability to enter, save, and execute a program written in C++
   c. Demonstrate the ability to apply good programming techniques

2. Learning Activities:
   a. Instructor will cover the syllabus. (C5, F1)
   b. The instructor will review the lab procedures for the microcomputer lab. (C5, C6, C8, C9, C19, F1, F5, F11)
   c. The student will write a program using the materials presented (C1, C5, C6, C16, C17, C18, C19, F1, F2, F3, F5, F9, F11, F13, F17)

3. Unit Outline: Follow the learning activities

B. Unit Two: Introduction to Classes and Objects

1. Learning Outcomes: Upon successful completion of this unit the student will be able to:
   a. Create user defined classes
   b. Create objects from user defined classes

2. Learning Activities:
   a. The instructor will demonstrate the design and creation of classes. (C5, C6, C8, F1, F3, F5, F11, F13)
   b. The instructor will demonstrate how to create and use objects from user defined classes. (C5, C6, C8, F1, F5, F11, F13)

3. Unit Outline:
   a. Follow the learning activities

B. Unit Three: Introduction to Classes and Objects

1. Learning Outcomes: Upon successful completion of this unit the student will be able to:
   a. Employ proper OOP principles in class designs
   b. Review for Mid-Term Exam

2. Learning Activities:
a. The instructor will discuss and demonstrate proper OOP design and analysis principles. (C5, C6, C8, F1, F5, F11, F13)
b. The students will write a program using the materials presented. (C1, C5, C6, C16, C17, C18, C19, F1, F2, F3, F5, F9, F11, F13, F17)

3. Unit Outline:
   a. Follow the learning activities
   b. Administer Mid-Term Exam

F. Unit Four: Pointers, arrays and vectors

1. Learning Outcomes: Upon successful completion of this unit the student will be able to:
   a. Demonstrate the use of pointers to point to addresses including arrays
   b. Demonstrate the use of arrays of objects
   c. Create arrays using the Standard Template Library Vector class

2. Learning Activities:
   a. The instructor will demonstrate the use of pointers to point to address. (C5, C6, C8, F1, F3, F5, F11, F13)
   b. The instructor will demonstrate how to create arrays of objects. (C5, C6, C8, F1, F5, F11, F13)
   c. The instructor will demonstrate how to create arrays using the Standard Template Library Vector class. (C5, C6, C8, F1, F5, F11, F13)
   d. The students will write a program using the materials presented. (C1, C5, C6, C16, C17, C18, C19, F1, F2, F3, F5, F9, F11, F13, F17)

3. Unit Outline: Follow the learning activities

G. Unit Five: Searching, Sorting, and Algorithm Analysis

1. Learning Outcomes: Upon successful completion of this unit the student will be able to:
   a. Demonstrate the use of various sorting routines
   b. Demonstrate the use of various search routines
   c. Determine the best algorithms to use through analysis

2. Learning Activities:
   a. The instructor will demonstrate linear and binary search routines. (C5, C6, C8, F1, F5, F11, F13)
   b. The instructor will demonstrate bubble and selection sort routines. (C5, C6, C8, F1, F5, F11, F13)
   c. The instructor will discuss algorithm analysis to determine the best algorithm to use in a particular situation. (C5, C6, C8, F1, F5, F11, F13)
d. The students will write a program using the materials presented. (C1, C5, C6, C16, C17, C18, C19, F1, F2, F3, F5, F9, F11, F13, F17)

3. **Unit Outline:**
   a. Follow the learning activities.
   b. Review for and administer Midterm Exam

H. **Unit Six:** Advanced Class design and Object-Oriented Programming

1. **Learning Outcomes:** Upon successful completion of this unit the student will be able to:
   a. Demonstrate the use of inheritance and aggregation in class hierarchy design
   b. Demonstrate the use of constructors and copy constructors in class design
   c. Demonstrate the use of overloaded operators when creating classes
   d. Demonstrate the use of class destructors
   e. Employ better usage of string class objects

2. **Learning Activities:**
   a. The instructor will demonstrate the use of inheritance and aggregation in class hierarchy design. (C5, C6, C8, F1, F5, F11, F13)
   b. The instructor will demonstrate the use of constructors and copy constructors in class design. (C5, C6, C8, F1, F5, F11, F13)
   c. The instructor will demonstrate the use of overloaded operators when creating classes. (C5, C6, C8, F1, F5, F11, F13)
   d. The instructor will demonstrate the use of destructors when creating classes. (C5, C6, C8, F1, F5, F11, F13)
   e. The instructor will demonstrate advanced usage of the C++ string class in program design. (C5, C6, C8, F1, F5, F11, F13)
   f. The students will write a program using the materials presented. (C1, C5, C6, C16, C17, C18, C19, F1, F2, F3, F5, F9, F11, F13, F17)

3. **Unit Outline:** Follow the learning activities

I. **Unit Seven:** Advanced File I/O, Polymorphism and Virtual Functions

1. **Learning Outcomes:** Upon successful completion of this unit the student will be able to:
   a. Demonstrate the use of sequential and random access techniques with file I/O
   b. Demonstrate the use of both text and binary file I/O
   c. Make abstract base classes
   d. Create pure virtual functions in class designs
2. **Learning Activities:**
   a. The instructor will demonstrate the use of sequential and random access techniques with file I/O. (C5, C6, C8, F1, F5, F11, F13)
   b. The instructor will demonstrate different techniques in text and binary file I/O. (C5, C6, C8, F1, F5, F11, F13)
   c. The instructor will demonstrate the use of abstract base classes and how to use them in polymorphic functions. (C5, C6, C8, F1, F5, F11, F13)
   d. The instructor will demonstrate how to create pure virtual functions in class designs. (C5, C6, C8, F1, F5, F11, F13)
   e. The students will write a program using the materials presented. (C1, C5, C6, C16, C17, C18, C19, F1, F2, F3, F5, F9, F11, F13, F17)

3. **Unit Outline:** Follow the learning activities

J. **Unit Eight: Exceptions, Templates, and the Standard Template Library (STL)**

1. **Learning Outcomes:** Upon successful completion of this unit the student will be able to:
   a. Demonstrate the use of error trapping techniques using exceptions
   b. Demonstrate the use of function and class templates to create generic code
   c. Demonstrate the use of the containers, iterators, and algorithms of the Standard Template Library (STL)
   d. Review for Final Exam

2. **Learning Activities:**
   a. The instructor will demonstrate the use error trapping techniques using exceptions. (C5, C6, C8, F1, F5, F11, F13)
   b. The instructor will demonstrate the use of function and class templates to create generic code. (C5, C6, C8, F1, F5, F11, F13)
   c. The instructor will demonstrate how to use containers, iterators, and algorithms of the Standard Template Library (STL). (C5, C6, C8, F1, F5, F11, F13)
   d. The students will write a program using the materials presented. (C1, C5, C6, C16, C17, C18, C19, F1, F2, F3, F5, F9, F11, F13, F17)

3. **Unit Outline:** Follow the learning activities
   a. Follow the learning activities.
   b. Review for and administer Final Exam