CENTRAL TEXAS COLLEGE
ITSE 1302
COMPUTER PROGRAMMING

Semester Hours Credit:   3

INSTRUCTOR:_____________________

OFFICE HOURS:__________________

I. INTRODUCTION

A. An introduction to computer programming including design, development, testing, implementation, and documentation. This course introduces the fundamental concepts of structured programming, and provides a comprehensive introduction to programming for computer science and technology majors. Topics include software development methodology, data types, control structures, functions, arrays, and the mechanics of running, testing, and debugging. This course assumes computer literacy. (This course is included in the Field of Study Curriculum for Computer Science.)

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. Prerequisite: COSC 1315 or ITSE 1329.

II. LEARNING OUTCOMES

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

A. Design, write, test, and document computer programs. (C1, C7, C8, C16, C18, F1, F3, F8)

B. Describe how data are represented, manipulated, and stored in a computer. (C11, C12, C13, C14, F1, F5, F9)

C. Categorize different programming languages and their uses. (C1, C7, C8, C16, C18, F1, F3, F8)
D. Understand and use the fundamental concepts of data types, structured programming, algorithmic design, and user interface design. (C5, C6, C8, F2, F11)
E. Demonstrate a fundamental understanding of software development methodologies, including modular design, pseudo code, flowcharting, structure charts, data types, control structures, functions, and arrays. (C1, C8, C16, C18, F1, F3, F8)
F. Develop projects that utilize logical algorithms from specifications and requirements statements. (C1, C7, C8, C16, C18, F1, F3, F8)
G. Demonstrate appropriate design, coding, testing, and documenting of computer programs that implement project specifications and requirements. (C5, C6, C8, F2, F11)
H. Apply computer programming concepts to new problems or situations. (C1, C8, C16, C18, F1, F3, F8)

III. INSTRUCTIONAL MATERIALS

A. The instructional materials identified for this course are viewable through www.ctcd.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>900-1000</td>
<td>A - Superior</td>
<td>4</td>
</tr>
<tr>
<td>Assessments</td>
<td>300</td>
<td>800 - 899</td>
<td>B - Above Average</td>
<td>3</td>
</tr>
<tr>
<td>Final Assessment</td>
<td>400</td>
<td>700 - 799</td>
<td>C - Average</td>
<td>2</td>
</tr>
<tr>
<td>TOTAL</td>
<td>1000</td>
<td>600 - 699</td>
<td>D – Passing but Unsatisfactory</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0 - 599</td>
<td>F - Failure</td>
<td>0</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](http://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. **Lesson Zero:** Course Orientation. Introduce course requirements and objectives. Introduce students to the laboratory and use of the microcomputer.

1. **Learning Outcomes:** Upon successful completion of this lesson the student will be able to: Discuss course requirements as defined in the syllabus and reviewed by the instructor

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)

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

B. **Lesson One:** Getting Started: Introduce how to program in C++.

1. **Learning Outcomes:** Upon successful completion of this unit the student will be able to:
   a. Enter, save, and execute a program written in C++
   b. Discuss good programming techniques

2. **Learning Activities:**
   a. The instructor demonstrates C++ programming. (C5, C6, C8, C9, C19, F1, F5, F11)
   b. 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. **Lesson Outline:** Follow the learning activities

C. **Lesson Two:** Expressions and Interactivity in C++

1. **Learning Outcomes:** Upon successful completion of this unit the student will be able to:
   a. Write arithmetic expressions
   b. Demonstrate the use of input procedures.
   c. Demonstrate the use of output procedures.
   d. Demonstrate the use of basic and string variables.
   e. Design, write, compile, and execute a C++ program using top-down designing techniques.

2. **Learning Activities:**
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. **Lesson Outline**: Follow the learning activities

D. **Lesson Three**: Making Decisions in C++

1. **Learning Outcomes**: Upon successful completion of this unit the student will be able to:
   a. Demonstrate the use of control flow (if-then, else).
   b. Demonstrate the use of relational expressions and logical operators.
   c. Control the flow of a program with decision statements.

2. **Learning Activities**:
   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. **Lesson Outline**: Follow the learning activities

E. **Lesson Four**: Looping with C++

1. **Learning Outcomes**: Upon successful completion of this unit the student will be able to:
   a. Demonstrate how to code effective looping constructs.
   b. Demonstrate the use of simple arrays.
   c. Demonstrate the use of counters and accumulators.
   d. Demonstrate the use of files for input and output.

2. **Learning Activities**:
   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. **Lesson Outline**: Follow the learning activities.

F. **Lesson Five**: Functions in C++

1. **Learning Outcomes**: Upon successful completion of this unit the student will be able to:
   a. Demonstrate the use of modularized programming techniques
   b. Create both void and value-returning functions
   c. Demonstrate how to pass parameters into functions
   d. Demonstrate an understanding of variable scope.
   e. Create overloaded functions.
2. **Learning Activities:**
   a. 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. **Lesson Outline:** Follow the learning activities.

---

G. **Lesson Six:** Arrays in C++

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:**
   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. **Lesson Outline:** Follow the learning activities

---

H. **Lesson Seven:** 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:**
   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. **Lesson Outline:** Follow the learning activities

---

I. **Lesson Eight:** Pointers

1. **Learning Outcomes:** Upon successful completion of this unit the student will be able to:
   a. Describe the relationship between arrays and pointers.
   b. Demonstrate the use of pointer arithmetic in programming code.
   c. Initialize pointers.
   d. Compare pointers in logical statements.
   e. Demonstrate the use of pointers as function parameters.
f. Demonstrate the use of pointers to constants and constant pointers.

2. Learning Activities:
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. Lesson Outline: Follow the learning activities