Warren County Community College

CSC 122 Programming II
Course Master Syllabus

Course Information

Catalog Description: This course addresses computer applications, standard algorithms, programming methodology, and applications. Files and records, pointer data types, elementary sorting and searching, and information storage and retrieval are covered.

Prerequisite(s): CSC 121 – Programming I

Corequisite(s): None

Credits: 3

Hours: 2 lecture and 2 lab

Required Text(s): MindTap Programming for Farrell's *Java Programming*

Supplemental Materials:

Instructor Information

Instructor's Name:

Contact Information:

Office Hours:

Program Goals

1. Demonstrate foundational knowledge of computer systems, including their components, functions, and applications.
2. Apply fundamental technology skills to solve problems and complete tasks in various computing environments.
3. Apply programming principles to solve problems using modern development tools.
4. Analyze and manage data communications and networking systems.

5. Apply critical thinking and data analysis techniques to solve computational problems and make informed decisions.
6. Design and develop interactive web content and apps using modern programming languages and development tools

Class Core Competencies

The following core competencies are embedded in this curriculum:

- Communicate effectively in both speech and writing
- Use scientific method of inquiry, through the acquisition of scientific knowledge
- Use computer systems or other appropriate forms of technology to achieve educational and personal goals
- Understand ethical issues and situations
- Address an information need by locating, evaluation and effectively using information.

Class Learning Assessment

Student Learning Outcomes	Suggested Means of Assessment
Describe and utilize functions, pointers, and control structures	Quizzes
Utilize and explain the use of classes, instantiation, methods, constructors, and destructors	Quizzes, lab assignments
Identify/describe the 4 principles of Object Oriented Programming; encapsulation, data abstraction, polymorphism and inheritance	Quizzes, lab assignments/projects
Utilize a text editor/IDE to write and compile advanced programs, including the use of object-oriented techniques to access and store files	Lab assignments/projects
Employ additional logic and control program flow	Lab assignments/projects

Student Learning Outcomes	Suggested Means of Assessment
Apply variables, communicate with different date types and create advanced arrays	Lab assignments/projects

Grading System

Letter Grade	Percentage Range
A	90 to 100 percent
B Plus (B+)	87 to 89.9 percent
B	80 to 86.9 percent
C Plus (C+)	77 to 79.9 percent
C	70 to 76.9 percent
D	60 to 69.9 percent
F	Below 60 percent

College Information

Disability Services
Disability Services Webpage

Statement and Policy on Cheating, Plagiarism, and Academic Conduct
Academic Conduct Policy (Policy 304) [PDF]

Students are required to perform all the work specified by the instructor and are responsible for the content and integrity of all academic work submitted. A violation of academic integrity will occur if a student: (1) knowingly represents work of others as one's own, (2) uses or obtains unauthorized assistance in any academic work, (3) gives fraudulent assistance to another student, or (4) furnishes false information or other misuse of college documents.

Unauthorized use of Artificial Intelligence (AI) is prohibited unless the instructor explicitly permits it in the syllabus or assignments. If tools such as ChatGPT are permitted by an instructor to be used for an assignment, they should be used with caution and with proper citation. In all such cases, you may only use the specific tools identified as acceptable and you must adhere to the College's standards for attribution, validation, and transparency.

In cases of suspected violation of academic integrity, the incident is to be reported to the Office of Academics. A student found guilty of violating the rule of academic integrity by the Vice President of Academics will be considered to have failed in personal obligation to the College; such failure will be subject to disciplinary action by the College. Unless otherwise notified, the instructor will allow students who are pending disciplinary action to attend class.

Tutoring and Testing at the Instructional Support Center (ISC)

- Instructional Support Center (ISC) Webpage

WCCC Library Services

- WCCC Library Website

Citation Styles and Formatting

Work submitted for any Warren County Community College course must conform to the required formatting and citation style that are specified by the instructor. The Library and Instructional Support Center can assist students with formatting and citations for assignments and papers.

WCCC Attendance Expectations

The College expects students to attend and participate in all classes. This involves sharing discussions and lectures in class, as well as interacting with other students. Grades are based on the quality of work students complete in meeting the requirements for a particular course, as stated in the course syllabus and catalog description. Attendance, among other factors, may be an additional criterion for assigning grades. Students should realize that they are responsible for all class meetings and that work missed because of absences may affect their grades. Attendance in online classes is determined by the completion of assignments/ tests. Students who have not completed any assignments or tests will be considered non-attending and receive an NF grade in online classes.

Class Attendance Expectations

Topical Outline

I. Creating Java Programs

a. Introduction to procedural and OO programming, classes, objects, inheritance and polymorphism
b. Saving, compiling, and viewing console output of Java program
c. Adding comments and organizing code
d. Introduction to GUI application

II. Using Data
a. Introduction to variable data types (string, boolean, floating-point, int)
b. Scanner and JOptionPane class for GUI
c. Java arithmetic
d. Data type conversion

III. Using Methods, Classes, and Objects
a. Creating, understanding, and using methods and classes
b. Declaring objects and using their methods
c. Introduction to using constructors

IV. More Object Concepts
a. Understanding blocks and scope
b. Overloading methods and constructors

V. Making Decisions
a. If/else blocks
b. Logic of AND, OR, NOT
c. Switch statements

VI. Looping
a. Common errors, arithmetic shortcuts and optimum performance using

VII. While and For loops
a. do… while loops
b. Nested loops

VIII. Characters, Strings, and the String Builder
a. Character and string methods
b. Declaring and Comparing String objects
c. Introduction to String Builder class

IX. Arrays
a. Declaring arrays
b. Searching arrays
c. Using arrays in loops and methods
d. Sorting arrays (simple sort, bubble sort, insertion sort)
e. Using multidimensional arrays

X. Inheritance
a. Using UML to visualize inheritance
b. Parent/child classes
c. Using constructors and methods during inheritance

d. Abstract Classes
e. Using Inheritance to achieve good software design

XI. Exception Handling
a. Recognize and fix problems with exceptions
b. Exceptions in methods
c. Creating an exception class

XII. File Input and Output
a. Understanding file system management and using pathnames
b. Using Java's IO class
c. Reading, computing, and then writing files

XIII. Swing Components
a. JFrame, JLabel, JButtons, JTextFields
b. Event driven programming

XIV. JavaFX and Scene Builder
a. Using Java for GUI applications
b. Connection to CSS
c. Animation using JavaFX

Grading Methods

Assignment	Points/Percentage

Class Itinerary

Week / Unit	Topic	Readings/Assignments

Week / Unit	Topic	Readings/Assignments