**C O U R S E O U T L I N E**

**CS 151 Computer Science I**

**Fall Semester, 2020**

2020/21 Catalog data: First course in Computer Science. Introduces the fundamental concepts of computer programming with an object-oriented language with an emphasis on analysis and design. Topics include data types, selection and iteration, instance variables and methods, arrays, files, and the mechanics of running, testing and debugging.

**Prerequisites:** MATH 152 (may be taken concurrently).

**Textbook**: John Lewis and William Loftus, Java Software Solutions, 9th edition, Pearson, 2017.

**Instructor**: Neli P. Zlatareva, Ph.D., Professor of Computer Science.

Office online

E-mail zlatareva@ccsu.edu

Web site http://www.cs.ccsu.edu/~neli/

Office hours M 5:00 - 5:55 pm, MW 12:05 -1:05 pm

TR 10:40 -- 11:15 am, 1:30 -- 2:00 pm

**Course Objectives**: Having completed this course successfully, the student should

- Understand the basic principles of object-oriented programming.

- Design and debug Java programs using basic control structures (linear, iteration, and selection).

- Define Java objects, methods and classes and use them in definite application settings.

- Understand and utilize arrays and text files.

- Learn how to solve non-trivial problems, and how to implement their solutions.

**NOTE: CLASS ATTENDANCE IS A MUST FOR SUCCESSFUL COMPLETION.**

**Class topical outline and assignments for the week:**

Week 1 August 26 - August 28

- Come prepared – the textbook is a must, as well have jdk downloaded – e-mail was sent!

- Introduction to computer systems and computer languages.

- Binary numbers.

- Assignments:

\*) Read Sections 1.1, 1.2, and 1.3.

\*\*) Install jdk on your home computer. Download from http://www.oracle.com/technetwork/java/javase/downloads/index-jsp-138363.html

Week 2 August 31 - September 4

- Introduction to Java language and Object-Oriented Programming.

- Writing and running Java programs. Comments, identifiers and reserved words.

- Character strings.

- Assignments:

\*) Read Sections 1.4, 1.5, and 1.6.

\*\*) Do Self-Review Questions from Chapter 1.

\*\*\*) Test your jdk installation by running provided on the course Web site

examples to ensure that it works properly.

\*\*\*\*) Review Lab 1 problems 1 and 2 (see course Web site).

Week 3 September 7 - September 11

- Primitive data types. Variables. The assignment statement and arithmetic operators.

- Data conversion.

- Interactive programs. The Scanner class.

- Assignments:

\*) Submit homework #1 (see deadline on the web site)

\*\*) Study Chapter 2 and do Self-Review questions from Chapter 2.

Week 4 September 14 - September 18

- Introduction to objects, classes, class libraries and packages.

- Formatting output.

- Wrapper classes.

- Increment and decrement operators.

- Boolean expressions.

- Assignments:

\*) Study Chapter 3.1, 3.2, 3.3, 3.5, 3.8

Note: GUIs will be covered in CS 152.

\*\*) Do Lab 2 problems 1, 2, and 3.

\*\*\*) Do Self-Review Questions from sections covered.

Week 5 September 21 - September 25

- Conditionals: if - else and switch statements.

- Loops: the while statement.

- Assignments:

\*) Do Lab 3, all problems – pay special attention to problem 2.

\*\*) Study Sections 5.1, 5.2, 5.3, 5.4, 6.1, 6.2.

Week 6 September 28 – October 2

- Loops: do and for statements.

- Nested loops.

- Assignments:

\*) Submit homework #2.

\*\*) Do Self-Review Questions from Chapters 5 and 6.

\*\*\*) Study Sections 6.3 and 6.4, and review Chapters 1, 2, 3, 5 and 6 (only sections covered).

Week 7 October 5 – October 9

- More about JAVA classes. Writing user-defined classes and methods.

- UML class diagrams.

- Encapsulation and visibility modifiers.

- Assignment: Review class-related sections from Chapters 2 and 3.

Week 8 October 12 – October 16

- Parameter passing.

- Dependencies among classes, and dependencies among objects of the same class.

- Assignments:

\*) Submit homework #3.

\*\*) Study Sections 4.1, 4.2, 4.3, 4.4, 4.5, and 7.4, 7.7 (parameter passing)

\*\*\*) Do Self-Review Questions from Chapter 4. Review Chapters 3, 4, 5 and 6 in preparation

for the midterm.

Week 9 October 19 – October 23

**- Midterm is to be held on October 20.**

- Static variables and static methods.

- Assignments:

\*) Study Sections 7.1, 7.2, 7.3

Week 10 October 26 – October 30

- Method overloading.

- The **this** reference.

- Interfaces.

- Assignments:

\*) Study Sections 7.4, 7.8

\*\*) Do Self-Review questions from Chapter 7.

Week 11 November 2 – November 6

- Enumerated types.

- Reading from a text file and writing to a text file.

- Assignments:

\*) Submit homework #4.

\*\*) Study Section 3.7, 7.6, 5.5

Week 12 November 9 – November 13

- Arrays of primitive types.

- Initializer lists.

- Assignment:

\*) Study Sections 8.1 and 8.2.

Week 13 November 16 – November 20

**Note: November 17 is the last day to withdraw from courses without permission**

- Arrays of objects.

- Command-line arguments.

- The ArrayList class.

- Assignment:

\*) Study Sections 5.6, 8.3 and 8.4

Week 14 November 23 – November 24 (Thanksgiving recess week)

- Dynamic arrays and multi-dimensional arrays.

- Assignment:

\*) Study Sections 5.6, 8.5 and 8.6.

\*\*) Do Self-Review Questions from Chapter 8.

Week 15 November 30 – December 4

- Review.

- Introduction to class hierarchies.

- Assignment:

\*) Submit homework #5.

\*\*) Prepare for the final exam.

**December 10: Final exam 9:25 am – 11:25 am**

**Tests and quizzes**: There will be one midterm and a final exam, and four pop-up programming quizzes (all to be given during synchronous classes). Test dates for the midterm and the final are stated in the course outline; quizzes may be given unannounced at any time. No make-ups for missed tests or quizzes will be granted unless due to exceptional circumstances (for the tests, must inform me BEFORE the test stars) supported by acceptable documentation. **NO EXTRA WORK TO COMPENSATE FOR POOR PERFORMANCE ON TESTS OR MISSED QUIZZES WILL BE GIVEN**, except for occasional preannounced extra work available to everyone willing to put an effort with PROVEN result (NOT THE EFFORT, BUT THE RESULT WILL BRING EXTRA POINTS). This extra work will add up to 5 points (5%) to your grade.

**Class discussion**: Since the style of the class sessions is interactive, students are expected to come to class with prepared questions, comments and answers to the assigned exercises and projects.

**Homeworks**: Five official homeworks will be given ("unofficial" ones need not be submitted -- but they are not less important -- do not ignore them). The student is responsible for constructing a set of test cases which cover all of the possibilities inherent in that program assignment. Homeworks will be submitted via Blackboard Learn. They must include the following: (1) a pseudo code OR UML chart (whichever is appropriate); (2) java code; (3) example runs (outputs) of the programs needed to demonstrate their correctness; and (4) all relevant files needed to run them. **Homeworks will be graded upon degree of success, precise implementation of the theoretical concepts** and use of an appropriate program structure as discussed in class. **No “**free **style” unrelated to the concepts discussed in class programs will be accepted – they will bring you 0 points even if they produce the expected result.** All homework assignments must be submitted by the due date. There will be a penalty for late submissions. Homeworks must be INDEPENDENT student work. “Share the work, share the grade” policy will be followed for grading shared work. Homeworks that are clearly NOT done by the student will be given 0 points.

**Academic honesty**: All homeworks, tests and quizzes must be an individual effort of the student submitting the work for grading. See the section "Policy on Academic Honesty" in the CCSU Student Handbook.

**Attendance**: It is expected that the student will attend class sessions regularly. Absences result in the student being totally responsible for the make-up process.

**Student conduct**: It is expected that all students will conduct themselves in a respectful manner, and will assist in maintaining an atmosphere conductive to learning in the classroom.

**Grades and evaluation**: The student will be evaluated regularly during the semester as follows

- Midterm: 20 points. It provides 20% of the final grade.

- Programming quizzes: 5 points each. They provide 20% of the final grade.

- Homework 1: 5 points. It provides 5% of the final grade.

- Homework 2: 6 points. It provides 6% of the final grade.

- Homework 3: 7 points. It provides 7% of the final grade.

- Homework 4: 8 points. It provides 8% of the final grade.

- Homework 5: 9 points. It provides 9% of the final grade.

- Final exam: 25 points. It provides 25% of the final grade.

The final grade for the course will be determined as follows:

Total points Final grade

--------------- --------------

93 and above A

90 - 92.99 A-

87 - 89.99 B+

83 - 86.99 B

80 - 82.99 B-

77 - 79.99 C+

73 - 76.99 C

70 - 72.99 C-

67 - 69.99 D+

63 - 66.99 D

60 - 62.99 D-

below 60 F