Spring 2021
CSCI-41
Programming Concepts and Methodology II
Syllabus

Instructor: Simon Sultana, Ph.D.
Department: Computer Science
E-mail: simon.sultana@reedleycollege.edu
Phone: (559) 638-0300 x3192
Response time: < 24 hours
Office Hours: Mon, Wed 11am-12pm, by appt
Section Number: 54096

Unit(s): 4
Dates: 1/11/2021 – 5/17/2021
Class Meeting: Mon & Wed 9-11am
Location: Online
Classroom: Zoom at https://cccconfer.zoom.us/j/93375914959?pwd=dzNSN0pPeXlxSmp4eGN1VXRBQnVHdz09

Prerequisites: CSCI-40 or ENGR-40.
Advisory: ENGL-125 – Writing Skills for College and ENGL-126 Reading Skills for College

Course Description: This course introduces application of software engineering techniques to the design and development of large programs, data abstraction and structures, and associated algorithms. Topics include linear and non-linear data structures such as lists, stacks, queues, trees, and graphs, algorithms for recursion, searching, sorting, and traversal.

Course Goals and Student Learning Outcomes:
Upon completion of this course, students will be able to:
- Analyze problems, design and develop computer programs to solve these problems.
- Construct classic data structures used in all computer programs.
- Use a software tool called a debugger to debug and test programs.
- Write programs using advanced programming concepts.

Objectives:
In the process of completing this course, students will:
- Write a total of 1000 to 2000 lines of program code.
- Write programs using object-oriented programming and the C++ language.
- Define and use dynamic arrays, linked list, stacks and queue data structures.
- Write programs using pointers, recursion and file manipulation techniques.
- Design, implement, and test the implementation of "is-a" relationships among objects using a class hierarchy and inheritance.
- Create proper test cases to test computer programs.
- Utilize binary search tree data structure.
- Use string, stack and queue classes defined in C++ Standard Libraries.
- Identify and correct syntax and logical errors in computer programs.

Student Learning Outcomes are statements about what the discipline faculty hope you will be able to do at the end of the course. This is NOT a guarantee: the ultimate responsibility for whether you will be able to do these things lies with you, the student. In addition, the assessment of Student Learning Outcomes is done by the department in order to evaluate the program as a whole, and not to evaluate individual faculty performance.

Required or Recommended Textbooks and Materials:

Textbooks:

1. **Required**: zyBooks, register and connect via Canvas (cost: $58).
2. **Optional:** *Starting out with C++ From Control Structures through Objects*, 9th Ed, By Tony Gaddis, Pearson

Additional Resources:

Towson University Cybersecurity Modules: https://cis1.towson.edu/~cssecinj/

Learning Management System: CANVAS:

Canvas (https://scccd.instructure.com/) is used to post announcements, course information, programming assignments, and grade. You will submit your programming assignments on Canvas.
To log-in Reedley College CANVAS:

Username: Your 7-digit student ID number.
Password: If you have not previously changed your password, it is:

First name initial (upper case) + *last name initial* (lowercase) + *date of birth* (mmddyy)
Example: John Smith born on July 9th of 1988 Password =Js070988

Computer Lab:

Onsite face-to-face meetings will take place in FEM 3 on campus. Microsoft Visual Studio C++ compiler is used for this class.

Topics:

A. Classes

1. Classes and Members; Friend Functions
2. Constructors and Destructors
3. Using a Namespace, Header file, and Implementation file
4. Operator Overloading

B. Inheritance, Polymorphism, and Virtual Functions

1. Introduction to Inheritance
2. Protected members and Class Access
3. Constructors and Destructors in Base and Derived Classes
4. Redefining Base Class Functions
5. Class Hierarchies
6. Polymorphism and Virtual Member Functions
7. Abstract Base Classes and Pure Virtual Functions

C. Pointers and Dynamic Memory

1. Pointers and arrays as parameters
2. A container class with a dynamic array
3. String class

D. Linked Lists

1. A fundamental node class for linked lists
2. A linked-list toolkit
3. A container class with a linked list

E. Software Development with Templates, Iterators, and the STL

1. Template functions
2. Template classes
3. Standard template classes and their iterators

F. Stacks
1. Stacks and the STL stack
2. Stack Applications

G. Queues
1. Queues and the STL queue
2. Queue Applications

H. Recursion
1. Recursive functions
2. Recursive Applications

I. Binary Search Trees
1. Binary trees
2. Binary tree representations
3. Binary tree nodes
4. Binary tree traversals
5. Binary search trees

J. Graphs
1. Graph Definition
2. Graph Implementation
3. Graph Traversal
4. Path Algorithm

K. Further topics
1. Hash Tables
2. Heaps
3. Sets
4. Secure programming

Tentative Schedule:

	Assignment Points	Date
Week 0: CSCI-40 Topics Review	**16**	**1/11/2021**
Class 9am online		1/11/2021
Class 9am online		1/13/2021
In-Class Activity	7	1/15/2021
zyBooks Participation Activities	9	1/17/2021
Week 1: More on Objects & Classes	**24**	**1/18/2021**
No classes (Martin Luther King, Jr. Day)		1/18/2021
Reading: Starting Out with C++ (SOWC++) Ch. 9.8, 9.10, 13.9, 14		1/20/2021
Class 9am online		1/20/2021
In-Class Activity	7	1/22/2021
zyBooks Participation Activities	7	1/22/2021
Schedule One-on-One	10	1/24/2021

Week 2: Inheritance & Abstract Classes	**54**	1/25/2021
Reading: SOWC++ Ch. 15		1/27/2021
Class 9am online		1/25/2021
zyBooks Participation Activities	7	1/29/2021
Class 9am online		1/27/2021
In-Class Activity	7	1/28/2021
Program #1 - C++ Fundamentals	40	1/31/2021
Week 3: Templates, Containers, & the STL	**57**	2/1/2021
Class 9am online		2/1/2021
Reading: SOWC++ Ch. 16.2-16.4, 17.1-17.5		2/3/2021
zyBooks Participation Activities	10	2/5/2021
Class 9am online		2/3/2021
In-Class Activity	7	2/4/2021
Program #2 - Inheritance	40	2/7/2021
Week 4: Intro to Data Structures & Algorithms	**15**	2/8/2021
Class 9am online		2/8/2021
Reading: SOWC++ Ch. 11, 17.6		2/10/2021
zyBooks Participation Activities	8	2/13/2021
Class 9am online		2/10/2021
In-Class Activity	7	2/11/2021
Week 5 : Exam #1	**100**	2/15/2021
No classes (Washington Day)		2/15/2021
Exam #1 (covers weeks 0-3)	60	2/17/2021
Program #3 - Container with Array	40	2/21/2021
Week 6: Searching & Algorithm Analysis	**54**	2/22/2021
Class 9am online		2/22/2021
Reading: SOWC++ Ch. 8.1, 8.2, Read Early Objects Ch. 9.6		2/24/2021
zyBooks Participation Activities	7	2/26/2021
Class 9am online		2/24/2021
In-Class Activity	7	2/25/2021
Program #4 - String Class Implementation	40	2/28/2021
Week 7: Sorting Algorithms	**17**	3/1/2021
Class 9am online		3/1/2021
Reading: SOWC++ Ch. 8.3, 8.4, 20.8		3/3/2021
zyBooks Participation Activities	10	3/5/2021
Class 9am online		3/3/2021
In-Class Activity	7	3/4/2021
Week 8: Linked Lists	**79**	3/8/2021
Class 9am online		3/8/2021
Reading: SOWC++ Ch. 18, 20.5		3/10/2021
zyBooks Participation Activities	12	3/12/2021
Class 9am online		3/10/2021
In-Class Activity	7	3/11/2021
Program #5: Recursion & Sorting	60	3/14/2021
Week 9: Stacks & Queues	**10**	3/15/2021
Class 9am online		3/15/2021

Reading: SOWC++ Ch. 19		3/17/2021
zyBooks Participation Activities	3	3/19/2021
Class 9am online		3/17/2021
In-Class Activity	7	3/18/2021
Week 10: Exam #2	**60**	3/22/2021
Class 9am online (Exam Q&A)	60	3/22/2021
Exam #2		3/24/2021
Week 11: Hash Tables	**53**	4/5/2021
Class 9am online		4/5/2021
Reading: Online resources (see Canvas)		4/7/2021
Class 9am online		4/7/2021
Program #6 - Linked List	40	4/11/2021
zyBooks Participation Activities	6	4/9/2021
In-Class Activity	7	4/8/2021
Week 12: Trees	**59**	4/12/2021
Class 9am online		4/12/2021
Reading: SOWC++ Ch. 21		4/14/2021
zyBooks Participation Activities	12	4/16/2021
Class 9am online		4/14/2021
In-Class Activity	7	4/15/2021
Program #7 - Stacks and Queues	40	4/25/2021
Week 13: Graphs	**22**	4/19/2021
Class 9am online		4/19/2021
Reading: Malik Ch. 20 or online resources (see Canvas)		4/21/2021
Class 9am online		4/21/2021
zyBooks Participation Activities	15	4/23/2021
In-Class Activity	7	4/22/2021
Week 14: Balanced Trees & Heaps	**59**	4/26/2021
Class 9am online		4/26/2021
Reading: Online resources (see Canvas)		4/28/2021
zyBooks Participation Activities	12	4/30/2021
Class 9am online		4/28/2021
In-Class Activity - Secure Programming	7	4/29/2021
Program #8 - BST tree	40	5/2/2021
Week 15: Exam #3	**60**	5/3/2021
Class 9am online (Exam Review)		5/3/2021
Exam #3	60	5/5/2021
Week 16: Sets & Optimal Algorithms	**74**	5/10/2021
Class 9am online		5/10/2021
Reading: SOWC++ Ch. 20 (skim)		5/12/2021
zyBooks Participation Activities	7	5/14/2021
Class 9am online		5/12/2021
In-Class Activity - Software Engineering Ethics	7	5/14/2021
Program #9 - Graphs	60	5/16/2021
Week 17: Finals Week	**187**	5/17/2021
Final Exam 9am online	150	5/17/2021

Professionalism	37	
TOTAL	1000	

Subject to Change:
This syllabus and schedule are subject to change. If you are absent from class meeting, it is your responsibility to check on any changes made while you were absent.

Evaluation:
Students will be evaluated on the basis of their performance on various assignments according to the following scale. The instructor reserves the right to adjust scores as it may be required throughout the semester.

Points in the course total 1000 and are distributed as follows:

zyBooks Participation Activities (15)	125
In-Class Activities (15, lowest dropped)	98
Programming Assignments (9)	400
Exams (4)	330
Professionalism/1:1	47

Final grade is assigned using following scale:

900-1000 points	A
800- 899 points	B
700- 799 points	C
600- 699 points	D
< 600 points	F

Attendance
Attendance for onsite and synchronous online meetings will be taken at beginning of each class. Students who leave before the end of class will be marked tardy. For synchronous online meetings you are expected to have your camera turned on and will have the best experience if you use earbuds/headphones. Please make sure to stay muted until you have a question or something to add so as to cut down on background noise.

Students will be dropped from the class if they fail to attend the first class session of the semester. During the semester up to final drop date, any student who missed two weeks of class meetings (cumulative) will be dropped from this class (i.e. 4 classes).

Make-up tests are limited to students who have made arrangements with the instructor **prior** to the announced testing date or those students who have been excused by High School Attendance Office. Exam material is constructed from class discussions, assigned readings, guest lectures, video presentations, and special assignments. **Unless the student receives prior approval from the instructor, no make-up tests will be allowed.**

Grading Policy
zyBooks Participation Activities:
ZyBooks participation activities must be completed before announced due date (typically at start of class on Wednesday for full credit, exception in the first two weeks). These activities allow you to actively engage each week's content. Points will be attributed according to the number of responses required and the percentage of the activities a student completes before class. Reference the following rubric:

Percentage	Before Wednesday Class	After Wednesday Class by

Completed	Points Awarded	End of Week Points Awarded
90%+	100% of points	90% of points
80%+	90% of points	80% of points
70%+	80% of points	70% of points
60%+	70% of points	60% of points
50%+	60% of points	50% of points
25%+	40% of points	30% of points
13%+	20% of points	10% of points

In-Class Activities:
In-class activities will be completed in groups of 2-3 during class meeting time with some outside of class time, as necessary. Only one assignment needs to be submitted per group. Be sure to include names of all students in the group or credit will not be given.

Programming Assignments:
Nine programming mini-projects are assigned in this course. Students will work individually and in groups of 2-3 (Program #5 and #8 only) and NOT share code with other individuals or groups. Projects are suggested to be turned in for grading by the due date indicated on the course calendar. However, these can be turned in up to one month later at no penalty (with the exception of Program #7 - #9 as all assignments are due before the final exam). Please be careful as it is not a good idea to get too far behind and the work will build up and you will likely have a difficult time succeeding in the class. Consideration is given to the following criteria: functionality, error-proofing/exception handling, efficient use of course concepts, documentation, and readability in addition to test case results. All programming exercises must include your name at the top of all programs submitted, or you will not receive credit.

Exams:
There are three midterm exams and one final exam. All will be timed. See course calendar for timing, content covered, and points allotted. Exams will consist of multiple-choice questions on syntax, program flow, etc. There will also be small programming problems. A more difficult late exam can only be arranged if you have an excuse verified by an impartial party (i.e., a doctor or a court note).

Late Policy
Points will be deducted for late homework problem sets at 5% per day late, up to one week late (not excepted more than one week late). Problem sets are to be submitted electronically in Canvas and all work must be shown for partial credit. Programming assignments can be turned in up to one month after the original announced due date, but no later than the Sunday before the week of final exams. POGIL exercises will not be accepted late as these are group endeavors.

College Policies:
Cheating & Plagiarism, see Cheating and Plagiarism under Campus Policies:

Cheating and plagiarism is prohibited in the class. Incidents of cheating and plagiarism will result a failing grade on the particular assignment in question. Please see Student Conduct Standards at https://www.reedleycollege.edu/about/about-us/policies-and-procedures/student%20conduct%20standards.html for more information about academic integrity.

Each student is expected to assist in the overall environment of the classroom making it conducive to learning.

Code created by external sources can be utilized with permission from the instructor and 1) must be properly documented so as not to plagiarize the work of others and 2) must not make up more than 10% of the actual code (exclusive of documentation) of the project.

It is expected each student will do his/her own work unless otherwise instructed. This course involves both individual work and collaborative work. It is your responsibility to understand the guidelines that apply to each kind of work, and to be clear about which assignments are individual assignments and which are collaborative.

Activities not designated as cooperative assignments in this course are to be done individually. Individual work is to be carried out entirely and solely by an individual. Programs will be checked for plagiarism using both computer and human similarity checkers. Take extreme precautions that your individual work is not viewed by other students. This includes deleting all your computer files from public workstations when you are finished, retaining private permissions on your files, destroying printouts of source code, and not letting other students use your personal computer where you store your coursework.

In addition, the work you submit must be entirely your original creation. Using solutions from any other source is forbidden; in particular, using solutions (either instructors' or other students') from previous offerings of this or other courses is not allowed. Using solutions found on the Internet or getting help from online forums is not allowed.

Assignments which appear to be the result of a "group effort", or appear to have been copied from another student, will be considered plagiarized. Violations of this policy will be reported as violation of academic integrity per college policy.

In-class activities are designated as pair or team activities. Students will be assigned a partner or will self-assign to complete these activities. Be sure to use these collaborative activities as a chance to master the skills, as there will be quizzes in which each person must perform the skills individually.

Accommodations for Students with Disabilities
If you have a verified need for an academic accommodation or materials in alternate media (i.e., Braille, large print, electronic text, etc.) per the Americans with Disabilities Act (ADA) or Section 504 of the Rehabilitation Act, please contact the Reedley College Disabled Students Programs & Services (DSP&S) Department at (559) 638-0332. You can find more information at https://www.reedleycollege.edu/student-services/disabled-student-programs-and-services/index.html.

Reedley College is committed to creating accessible learning environments consistent with federal and state law. To obtain academic adjustments or auxiliary aids, students must be registered with the DSP&S office on campus. DSP&S can be reached at (559) 638-3332. If you are already registered with the DSP&S office, please provide your Notice of Accommodation form as soon as possible.

Important College Dates Spring 2021

Class begin	Monday	01/11/2021
Last day to drop a full-term class for a full refund	Friday	01/22/2021
Last day to register	Friday	01/29/2021
Last day to drop this class to avoid a “W” in person	Friday	01/29/2021
Last day to drop this class to avoid a “W” on Web	Sunday	01/31/2021
Last date to drop this class	Friday	03/12/2021
No classes, campus is closed		
Martin Luther King, Jr. Day	Monday	01/18/2021
Lincoln Day	Friday	02/12/2021
Washington Day	Monday	02/15/2021

Spring Recess	Mon-Fri	03/29 – 04/02/2021
Final Exam	**Monday**	**05/17/2021**