



CSCI 170-01 Software Design Principles

COURSE SYLLABUS: Fall 2019

INSTRUCTOR INFORMATION

Instructor: Dr. Donghwoon Kwon

Office Location: Starr 302D

Office Hours: Mon through Fri 11:00AM-11:50AM & Wed 1:00PM-1:50PM

Office Phone: 815-226-4056

Office Fax: 815-394-5166

University Email Address: dkwon@rockford.edu

Preferred Form of Communication: Email

Communication Response Time: Within 24 hours

COURSE INFORMATION

Course Description

The first course in Computer Science to be taken before any other CSCI major course. The focus is on software design techniques. The student is introduced to a disciplined approach to problem-solving and algorithm development, to an introduction to procedural and data abstraction. The course introduces the use of high-level programming language as a problem-solving tool, including basic data structures and algorithms, structured programming techniques, and software documentation. Scheduled: Yearly, Meets: MNO

Course Objectives

- To introduce students to a disciplined approach to problem-solving and algorithm development, with an introduction to procedural and data abstraction.
- To introduce students to the fundamentals of Computer Science.

Credit Hours: 3

The syllabus is subject to change.

Prerequisite: None

Class Type: Lecture & Lab

Class Hours: Mon, Wed, and Fri 10:00AM – 10:50AM at Starr Science Hall, Room 108C

Textbook(s) Required

- **Starting Out with Programming Logic and Design 4th edition** by Tony Gaddis, 2015, Pearson, ISBN10: 0-13-398507-5, ISBN13: 978-0-13-398507-8

Supplemental Textbook(s)

- **Starting Out with Python, 4th Edition** by Tony Gaddis, 2018, Pearson, ISBN 10: 0134444329, ISBN 13: 978-0134444321
- Instructor's own materials will be regularly and electronically provided.

Software Required

- *C++ programming for Windows users:* Microsoft Visual Studio in the CSCI lab or DevC++ for Windows users. You can download each software from <https://visualstudio.microsoft.com/vs/express/> or <https://sourceforge.net/projects/orwelldevcpp/files/latest/download>.
- *C++ programming for Mac users:* Xcode.
- *Python for Windows users:* Jupyter notebook on the Anaconda platform for Windows users.
- *Python for Mac users:* Canopy Express from <https://assets.enthought.com/downloads/#default> or students can choose any other software for Python coding.
- Please keep in mind that you should install Python version 3.0 or above. Python version 2.x is not allowed.
- If you have a difficulty to download and install software mentioned above, please contact the instructor.

Optional Texts and/or Materials

- Flash drive recommended

Student Learning Outcomes

Students will be able to

1. Show how computer hardware represents information.
2. Explain how computing components may be combined to build computer systems.
4. Apply general problem-solving strategies to the development of computer algorithms.
5. Write programs in high-level languages to express and implement algorithms to solve problems.
6. Identify and explain the application of abstract data.
7. Utilize information system software to organize, manipulate, and secure data.
8. Apply the object-oriented methodology to computer problem solving.

Topical Outline

- Topic 1: Introduction to Computers and Programming
- Topic 2: Input, Processing, and Output

The syllabus is subject to change.

- Topic 3: Flowchart & UML Diagrams
- Topic 4: Boolean Logic
- Topic 5: Programming Structure (Local and global variables, functions, etc.)
- Topic 6: Loop Statement
- Topic 7: Data Structures: Arrays
- Topic 8: Data Structures: Sorting
- Topic 9: Data Structures: Searching
- Topic 10: Pointer
- Topic 11: File Input / Output
- Topic 12: Menu-Driven Programs
- Topic 13: Function Point Analysis (FPA)
- Topic 14: Structure and Objected Oriented Programming (OOP)

COURSE REQUIREMENTS

Instructional Methods

1. Lectures: Important materials from the text and outside sources will be covered in class. Students should plan to take careful notes as not all materials can be found in the texts or readings.
2. Labs: Based on lectures, lab exercises will be given to students. Students are required to complete lab exercises, and the discussion is highly encouraged.
3. Assignments: Programming and / or general assignments will be regularly given to students.
4. Quizzes: Occasional announced quizzes will be given to help ensure students keep up with assigned materials.
5. Exams: Two exams will be given, one midterm exam and one final exam. Midterm exam will primarily cover topics from week 1-7, and final exam will be from week 8-the last week of class.

GRADING

Final grades in this course will be based on the following scale:

% of Total Points	Grade
87% - 100%	A
77% - 86.9 %	B
67% - 76.9%	C
57% - 66.9%	D
Below 57%	F

Category	Percentage
Midterm	20%
Final Exam	30%
Assignments	20%
Quizzes	15%
Attendance	15%

COURSE SPECIFIC PROCEDURE / POLICY

1. **Assignments:** All assignments **MUST** be turned in by the assigned deadlines. All assignments are due at the time specified. Please keep in mind that no late work will be accepted without penalty. If an assignment is turned in after the due date, **20%** of the grade will be forfeited each day. No assignment will be graded if submitted **5 or more days** after it is due. An assignment must be submitted within 5 days of the due date if you want it graded. All assignments must be placed in the appropriate Dropbox on Canvas.

The syllabus is subject to change.

2. Examination Makeup Policy: If a student is absent from an exam during the scheduled time for that exam, the student will automatically receive a grade of 0 for the exam unless:

- a. the student notifies the instructor of the absence before 24 hrs of the exam and supplies a written doctor's excuse or any other official documents explaining the absence, or
- b. there is an extraordinary situation which the instructor allows as an acceptable excuse (instructor needs to be notified before 24 hrs of the exam). If (a) or (b) applies, arrangements for a makeup exam will be made.

It will be the responsibility of the student to show written documentation supporting the absence, from your team coach, physician, or other relevant authority.

COLLEGE POLICY

1. Attendance: Each student is required to be present at all class lectures and labs. If an unforeseen absence does occur, the student is responsible to get the notes and assignments from another student. If a student has more than 3 absences from class without acceptable reasons, they may be dropped from the class roster.

This 3-credit course will meet for 150 minutes per session one time a week throughout the semester. A minimum of 2-3 hours of student preparation time outside of class is expected for each credit hour. Thus, please be prepared to devote 9-12 hours per week to this course.

2. Instructor Policy: Students not regularly attending class or not turning in assignments will be given a grade of “F” at the end of semester if that student has not dropped the class or been dropped by the instructor.

3. In case of an emergency or extenuating circumstances, such as illness, family crisis, contact me or have someone on your behalf contact me immediately; emergencies will be handled on a case-by-case basis. Email is best method for contacting me.

HONOR CODE

In this course the policies and procedures concerning the Rockford University Academic Honor Code including definitions of cheating and plagiarism as they appear on the appropriate pages of the current Rockford University Handbook will be Applicable.

PLAGIARISM POLICY

1. Plagiarism: To plagiarize is to present someone else's ideas or work as your own. Credit (citation) should be given to the source in the following instances: (1) when you directly quote someone else; (2) when you use someone else's ideas or opinions (unless they are common knowledge); (3) when you use someone else's examples; (4) when you cite statistics or other facts compiled by someone else; (5) when you present evidence or testimony taken from someone else's argument (Berke, Jacqueline. *Twenty Questions for the Writer*. 4th Ed. New York: Harcourt, Brace, Jovanovich, 1985).

If a student plagiarizes, that student will receive an “F” for the assignment. A second occurrence of plagiarism will result in expulsion from the course.

2. Copy: Copying parts or whole of assignments, quizzes and exams is just as serious as any other type of plagiarism. Any indication of copying, cheating and/or plagiarism on an exam/assignment/project will be an automatic 0 (zero) for the exam/assignment/project for all students involved.

ADA STATEMENT

Students with Disabilities: If you believe you are eligible to receive any type of academic accommodation, through such federal laws as the ADA, please contact the Lang Center for Health, Wellness, Counseling and Disabilities Services, 815-226-4083. The Lang staff manages disability services for Rockford University.

ACADEMIC CONCERN WITH THIS COURSE

A student who questions the justice of a final grade must first seek an explanation from the course instructor. If dissatisfied with the explanation offered, the student may appeal the grade. Additional information regarding grade appeals can be found in the Academic Catalog.

ELECTRONIC DEVICES POLICY

Electronic devices may be used in the classroom as long as they are being used for academic purposes as approved by instructor and/or are an approved accommodation for a documented disability.

DISCLAIMER

- Due dates, assignments, etc. are subject to change as directed by your instructor during the course of the semester.
- If you have questions about Computer Science, computer careers, etc. please email Dr. Donghwoon Kwon at dkwon@rockford.edu

COURSE OUTLINE / CALENDAR

Week	Lectures	Topics
1 (8/21 - 8/25)	Lecture 1	Course introduction and syllabus discussion Introduction to computers and programming
	Assignment #1: Binary values and number systems Due date: By 11:59PM, 9/1/2019	
2 (8/26 – 9/1)	Lecture 1	Introduction to computers and programming (Cont’)
	Assignment #2: Programming for data representation Due date: By 11:59PM, 9/1/2019	
3 (9/2 – 9/8)	Lecture 2	Input, processing and output with flowcharts
	Assignment #3: Programming for user inputs and outputs Due date: By 11:59PM, 9/8/2019	

The syllabus is subject to change.

4 (9/9 – 9/15)	Lecture 3	Variables and constants definition and math operations Decision structures based on if~else statements and Boolean logic
	Assignment #4: If, else, and else if statements with Boolean logic Due date: By 11:59PM, 9/15/2019	
5 (9/16 – 9/22)	Lecture 4	Break, continue, and switch~case statements Repetition structures (loops)
	Assignment #5: Loops programming Due date: By 11:59PM, 9/22/2019	
6 (9/23 – 9/29)	Lecture 5	Functions
	Assignment #6: Functions programming Due date: By 11:59PM, 9/29/2019	
7 (9/30 – 10/6)	Lecture 6	1 dimensional arrays Multi-dimensional arrays
	Assignment #7: Arrays programming Due date: By 11:59PM, 10/6/2019	
8 & 9 (10/7 – 10/20)	Lecture 7	Sorting algorithms, Midterm review session, <u>Midterm</u> , and <u>Fall break</u>
10 (10/21 – 10/27)	Lecture 8	Searching algorithms
	Assignment #8: Sorting and searching Due date: By 11:59PM, 10/27/2019	
11 (10/28 – 11/3)	Lecture 9	Pointers
	Assignment #9: Pointer for swapping numbers Due date: By 11:59PM, 11/3/2019	
12 (11/4 – 11/10)	Lecture 10	Files Menu-Driven Programs
	Assignment #10: Make menu-driven program Due date: By 11:59PM, 11/10/2019	
13 (11/11 – 11/17)	Lecture 11	Unified Modeling Language (UML) Graphic User Interface (GUI)
	Assignment #11: Draw UML diagrams and mock-up design Due date: 11:59PM, 11/17/2019	
14 (11/18 – 11/24)	Lecture 12	Function Point Analysis (FPA)
	Assignment #12: FPA counting Due date: 11:59PM, 11/24/2019	
15 (11/25 – 12/1)	Lecture 13	Structure and brief concept of Objected Oriented Programming (OOP)
	<u>Thanksgiving break</u>	
16 (12/2 – 12/8)	Lecture 14	Structure and brief concept of Objected Oriented Programming (OOP) (Cont')
	-	Review session for the final exam
17 (12/9 – 12/15)	<u>Final Exam</u>	

Date Prepared: August 19, 2019

The syllabus is subject to change.