CSE 381 SYLLABUS AND PORTFOLIO

This course introduces formal techniques to support the design and analysis of algorithms, focusing on both the underlying mathematical theory and practical considerations of efficiency. Topics include asymptotic complexity bounds, techniques of analysis, and algorithmic strategies.

In other words, it's a blast!!

1. Objectives

- Analyze and compare computer algorithms using: big-Oh, big-Omega, big-Theta, and Recurrence Relations
- Describe and implement in a high-level language the following algorithmic techniques: Divide and Conquer, Greedy, Dynamic Programming, Backtracking, and Branch and Bound

2. Prerequisites

You must have successfully completed the following courses:

- CSE 280 Discrete Mathematics
- CSE 121a Clojure Language

You must have *some* working knowledge of:

- Procedural, Object-Oriented, and Functional Programming
- Basic data structures (sets, lists, maps, trees, graphs, etc.)
- Summation notation (Σ)
- Recurrence relations
- Limits
- Logarithms
- Matrices
- Proofs

3. Requirements

You are required to obtain

3.1. **Text. Introduction to the Design and Analysis of Algorithms**.*Anany Levitin*,Third Edition, 2012, Pearson. (ISBN: 9780132316811)

3.2. **Documents.** As provided by the instructor.

3.3. **Software.** A text editor of your choice that can produce pdf files from LaTeX.

Date: May 12, 2022.

4. Behavioral Requirements

You are required to...

- attend class, as assessments will happen in class each day that are not reproducible outside of class.
- read assigned portions of the course materials *before* the class each Monday.
- complete all team and personal assessments to deepen your understanding of selected topics.

5. Course Periodicity

This course has a weekly period, i.e. you can count on knowing ahead of time what you will be doing each day of each week. Each class period consists of three 20 minute sections. On the first day of class these sections are:

(1) ***Presentation***
A time where I will add depth information to the preparation material you've read **before class**.
(2) ***Class Directed Learning***
You will participate in a class-wide activity that reinforces what you've read and what I've shown you.
(3) ***Create and Explain Solutions to Exercises***
During this time period, as a team of 2 or 3 you will create a solution to an exercise of your choice from the list of exercises for this week. On completion of your exercise you, as an individual, will explain your solution to someone not on your team until they understand your solution.

On the second day of class, the three sections are:

(1) ***Answer Questions***
I will answer questions that have been submitted to the course's channel.
(2) ***Class Directed Learning***
You will participate in a class-wide activity that reinforces what you've read and what I've shown you.
(3) ***Work Problems***
This is in-class time for individual work on the problem set for the week. Successful students will have started working on the problem set **before** this half-hour.

5.1. Questions.

The questions answered on Wednesdays are generalized from those you submit via the course's channel on Monday evenings. You must submit any and all unanswered questions on Monday evening. Not submitting questions leads to a reduced learning experience. You will have plenty of questions. Submit them! Choose knowledge not ignorance.

5.2. **Exercises.** Exercises are smaller experiences that are designed to float uncertainties and questions you have to the surface of your mind. They are designed to be smaller so you can find out what you don't know and then take the steps necessary to know.

5.3. **Problems.** Problems are weightier experiences that invite you to explore topics in algorithms and complexity, as well as increase your algorithmic problem solving prowess. All involve writing mathematically.

5.4. **Obtaining the Problems and Exercises.** These are found in the iLearn module for each week.

6. Assessment

Quatri-weekly, every four weeks, you will meet with me in my office, or via Zoom if you are part of a remote class. The purpose of this meeting is for you to present your portfolio of work to me, make a grade-to-date claim, and provide evidence regarding why that grade is correct. Your portfolio **MUST** follow the example portfolio's format and be complete and internally consistent. You are required to produce the portfolio using LaTeX, but can use any text editor of your choice (Yes, you could use VS Code with a plugin like LaTeX Workshop or any other LaTeX editor you choose. No...Word, Pages, and other non-text editors would not be good choices). You can also choose to use an online LaTeX editor if you would prefer.

The portfolio you bring to me for our meeting must be a hard-copy of the pdf generated from your LaTeX file if this is a face-to-face class, or be a displayed pdf file if the class is a remote class.

6.1. **Late Work.** Late work is accepted *only if* the reason is extraordinary, and acceptance is reached through private and prolonged negotiation. Also, you must come talk to me in person in my office (or in a zoom meeting if remote) — NOT by email, nor any other means of communication.

6.2. **Grades.** In each of our three personal meetings, you will present your portfolio and a letter based grade-to-date claim. Afterwards I will give you my thoughts on the strength of your claim. The last claim that you make, taking into account any feedback from me, will be your final grade for the course. All of your claims must must be evidence based. That means you must bring the evidence with you, in your portfolio, that supports your claim.

6.3. **Letter-Based-Grades.** You are required to use the definition of the grades from the University Catalog:

- **A** represents outstanding understanding, application, and integration of subject material and extensive evidence of original thinking, skillful use of concepts, and ability to analyze and solve complex problems. Demonstrates diligent application of Learning Model principles, including initiative in serving other students. Note: Notice this description of A implies that you have gone above and beyond. To claim this grade, throughout the 4 week period being reviewed, you *must* have consistantly done things similar to what you see in the list below and recorded evidence of this behavior in your portfolio. Examples of the required types of behaviors are:

- teaching and/or helping others in the class but not in your group,
- helping a Non-CS, Non-CE, Non-EE, Non-SE major with their homework for a non-tech class using the principles you learned in this class,
- applying what you've learned in this class in another class you are currently taking, and
- doing work not assigned such as writing code using what you are learning that has not been assigned, etc.

B represents considerable/significant understanding, application, and incorporation of the material which would prepare a student to be successful in next level courses, graduate school or employment. The student participates in the Learning Model as applied in the course.

C represents sufficient understanding of subject matter. The student demonstrates minimal initiative to be prepared for class. Sequenced courses could be attempted, but mastering new materials might prove challenging. The student participates only marginally in the Learning Model.

D represents poor performance and initiative to learn and understand and apply course materials. Retaking a course or remediation may be necessary to prepare for additional instruction in this subject matter.

7. University Policies

To review University policies regarding disabilities, sexual harassment, etc., or to arrange for a tutor from the Academic Support Center, select **Modules** in the iLearn course, scroll to the Student Resources module, and select the appropriate link.

8. Other

This document may be modified by the instructor at any time without notification.

9. Readings

These readings are to be completed *prior to* each listed week's Monday class.

10. Portfolio

10.1. **Course Tracker.** You are required to track your progress through the course using this table.

Note: Currently, you see full credit for week one's work. (✓means yes. Blank means no.) Use what you see in the LaTeX for week one in the table below to update the table for week 1 and all subsequent weeks each class day and week during the semester.

Course Tracker								
Week	Monday					Wednesday		Friday
	CRU	PFP	CDL	CAE	SAQ	PAQ	CDL	PPL
1	✓	✓	✓	✓	✓	✓	✓	100%
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								

This is an honest and true record of my work for this course.
Signature:____
Your Name Here

10.1.1. *Tracker Acronym Key.* Course Tracker acronyms and their meanings.

- **CRU** - I Completed the Reading and achieved a level of Understanding **before** the start of Monday's class and recorded questions about the items I didn't understand.
- **PFP** - I was present for and attentive to the presentation for this date.
- **CDL** - I fully participated in the Class Defined Learning for this date.
- **CAE** - I fully participated in the Create And Explain portion of the class for this date.
- **SAQ** - I submitted *at least 1* appropriate, Significant, Actual Question I have regarding the information for this week.
- **PAQ** - I was Present for and Attentive to the Answer Questions presentation for this date.

- **PPL** - I, individually, correctly completed this Percentage of the Problems and exercises showing this Level of understanding before Friday at Midnight.

10.2. **Grade Claims.** On the week indicated, bring this updated document to my office and make your claim.

Claim Week	Grade Claim	Instructor Grade	Adjusted Grade
5			
9			
13 - 14			

10.3. **Evidences.**

10.3.1. *Week 1.*

(1) Some Exercise or Problem Description.
Solution

(2) Some Other Exercise or Problem Description.
Solution requiring Code

```
(defn factorial
  ([n]                       ; when only one argument is passed in
    (factorial n 1))
  ([n acc]                   ; when two arguments are passed in
    (if  (= n 0)  acc
    (recur (dec n) (* acc n)))))
```