Data Structure Lab 5
Spring 2021 Wed, April 28

Lab Goals:

- Project Idea Approval (Remaining students)
- Project Screens Finalization (Discuss and Improve)
- Database Design for Project & File Handling
- Pre-lecture Exercises Complete Understanding (Next Week)
- Python notebooks Complete and Thorough Understanding (Next Week)
- Homework Questions

Project Idea Approval:

All the remaining students should get their project ideas approved in this lab.

Project Screens Finalization:

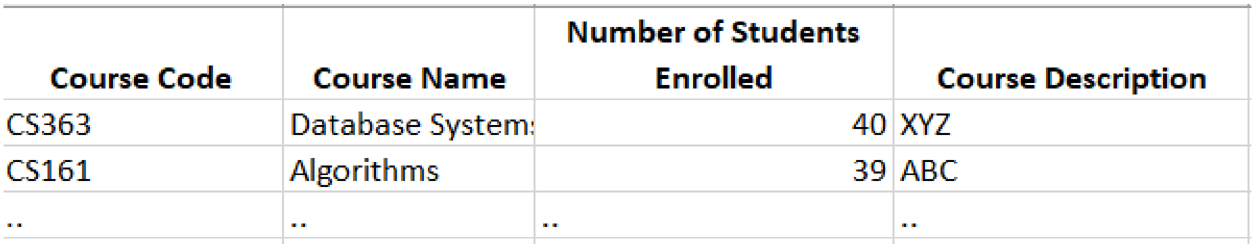
Discuss your screens and project flow with your me/TAs, get feedback and improve them for next time.

Database Design and File Handling:

For a working project, database is must. It is a storage structure where you keep all of your items which you want to store permanently. For example, if you want to implement a LOGIN feature in your project then you must check user's email and password with the email and password he has used to registered earlier. Where will we get his registration email and password? If we will store it in a variable and restart the program, then it will be lost. For this, we need to save it in our database. Next time, when user tries to login, we will fetch information from our database, compare it with the user's written email and password information and then decide if we need to take him to the next screen or not. For our case, our database is files like notepad or excel sheets or csvs etc. We will study more about databases in CS363 course.

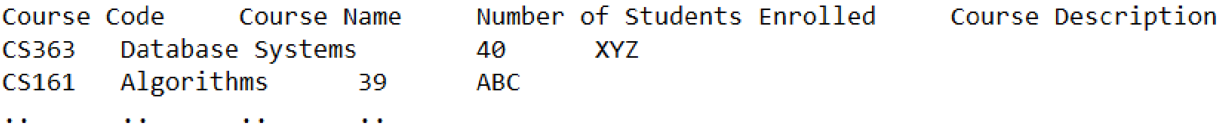
In this section, you will decide entities and their relations for your project. For example, if you are making a Course Management System and you decide that your entities are Course, Students and Enrollment etc. Once, your entities are defined then you can decide for their attributes. These attributes will serve as columns of files, just like in excel sheet or notepad file. For example, a course can have attributes like code, name, description and number of students enrolled etc. You can save your data in excel or csv file and write code in your project to fetch entries from and save entries in excel. Similarly, you can use notepad file as well where you can add entries in each line and separate their attributes by a comma, tab space or any token. Three versions are given below for your guidance.

Version1:



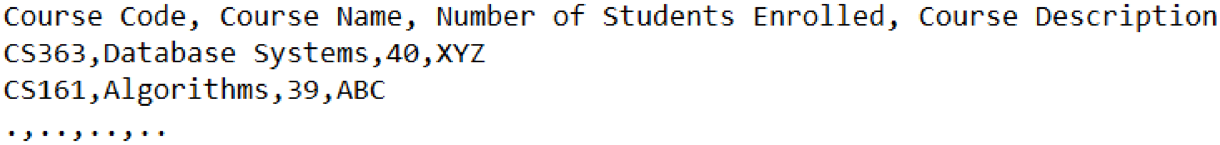
Course Code	Course Name	Number of Students Enrolled	Course Description
CS363	Database System:	40	XYZ
CS161	Algorithms	39	ABC
..	..	..	..

Version2:



```
Course Code      Course Name      Number of Students Enrolled      Course Description
CS363    Database Systems         40       XYZ
CS161    Algorithms       39       ABC
..       ..       ..       ..
```

Version3:



```
Course Code, Course Name, Number of Students Enrolled, Course Description
CS363,Database Systems,40,XYZ
CS161,Algorithms,39,ABC
.,..,..,..
```

Pre-lecture Exercises

In this pre-lecture exercise, we will remember a little bit of probability!

1. Let X be a random variable which is 1 with probability 1/100 and 0 with probability 99/100.

(a) What is the expected value E[X]?

(b) Suppose you draw independent random variables, X1, X2, . . . , Xn, distributed like X. What is the expected value $\mathbb{E}\left[\sum_{i=1}^{n} X_i\right]$?

(c) Suppose we draw independent random variables X1, X2, . . . and we stop when we see the first 1. For example, if we draw

X1 = 0, X2 = 0, X3 = 0, X4 = 1, . . .

then we would stop at X4. Let N be the last index that we draw; so in the previous example, N = 4. How big do you expect N to be?

Note: Actually, figuring out E[N] from scratch is a bit tricky. But even if you don't do it rigorously, intuitively how big do you expect N to be?

2. Consider the following pseudocode, which sorts an array A. Someone else has implemented the random shuffling for us, so do not worry about the details of that procedure.

```
BogoSort(A) :

while A is not sorted do

    randomly shuffle A

return A
```

(a) Let Xi be a random variable which is 1 if A is sorted after the i-th call, and 0 otherwise.

(b) What is E[Xi]?

(c) What is the expected number of times that BogoSort executes the while loop?

Python Notebooks:
Solve and thoroughly understand Python notebook provided in this lab.

Homework Question:

1. [**Majority Element**] Suppose we are given an array A of length n with the promise that there exists a majority element (i.e. an element that appears > n / 2 times). Additionally, we are only allowed to check whether two elements are equal (no > or < comparisons). Design an O(nlog(n)) algorithm to find the majority element, using divide and conquer. Informally, explain the correctness and runtime of your algorithm. [We are expecting actual code, an English description of the main idea of the algorithm, as well as an informal explanation of correctness and runtime]
2. [**Median of Two**] Given two arrays of length n, find the median of all elements of the two arrays.
   (a) If the arrays are unsorted, what is the best you can do?
   [**We are expecting**: A runtime and a brief description of your algorithm and code]
   (b) If the arrays are sorted, can you do better?
   [**We are expecting:** A short English description, code, runtime analysis.]
3. Implement **QuickSort** and compare it with Merge and Insertion Sort with small, large, very large and extremely large arrays (just like you did in Lab1)