COP 3502 – Section 3

Midterm Exam Overview

Date, Time and grade

- Date of Exam: Monday, October 19, 2020
- The exam will be available from 7:20 pm to 8:55 pm (starts 10 minutes before your regular class start time and ends 10 minutes after your class start time)
- It will be **timed** 1 hour and 20 minutes exam
- Exam will be auto submitted at 8:55 pm (if you start late)
- Total grade for the midterm written exam: 100 *(It will cover 20% of your course grade)*

Exam Environment

- **Exam will take place in Online Lock Down browser with webcam setting.**
- **Make sure your browser is installed from the link in the test Lock Down browser and your webcam is working.**
  - **Otherwise it will not work if you download it from other places**
  - **You must have to test your browser before the exam. Otherwise, no makeup will be arranged in case of technical difficulties during the exam if I see that you were neglecting to test the browser before the exam.**
- The exam will be mix of free text, fill in the blanks, matching, multiple choice, and coding questions.
  - But you need to use scratch paper to get many of your answers.
  - Use of compiler is not allowed
  - You cannot use your own calculator. (Use the calculator provided by lock down browser)
- Note that webcourses free text field, does not support **tab key**.
  - So, try to use space key to make the code more readable. But do not spend too much time on formatting if you run out of time.
  - Also, there is an option to use pre-formatted text option (instead of using paragraph). Try using that!
- Caution!
  - It is an individual close book exam.
  - You are not allowed to use phone, take picture of the question, Share the question partly or fully. This is also not a group exam .
  - Such activities will be considered as dishonesty and cheating!
  - In case you need any clarification, you can use your phone for just for emailing to me. We can cross check your recorded exam with your phone checking behavior, time stamp when you have used your phone and when you have sent us any email for any clarification.

Topics for the midterm exam

- C review such as pointer, pointer arithmetic, function call by reference and values, Structure, String operation, 2D array, File I/O
- Dynamic memory allocation
- Base conversion
- Abstract data types Stacks and applications
- Queues
- Linked list
- Recursion
- SLMP and Binary Search
- Algorithm Analysis (1, 2, summation, and 3)
- Algorithm analysis exercise
- Recurrence relation and exercises.

Question Pattern (read it and practice accordingly!)

- **You got some ideas about pattern of questions from the quizzes, but as the exam has more times and topics, there will be many questions with various level of difficulties.**
- **There will be enough time to answer them all if you do not mess-up much.**
- **Question can be:**
- **Multiple choice questions/True False/fill in the blanks**
- **Base conversion**
  - **Converting one number system to another. You will have to show steps of the conversion.**
- **Simulation question/ showing steps**
  - **Example: Converting a given infix to postfix and evaluation (if there is any question on it, you have to show the steps like the lecture slide), what is the status of the stack on a given situation, what is the status of a given circular queue if you delete something, or remove something (example rear or front values, is it full or empty), show the particular steps of recursion like tower of Hanoi, Fibonacci series, etc.**
- **Writing block of code *or writing a function* or *writing a recursive function* to perform a particular operation.**
- **Example question:**
  - **Given a linked list/doubly linked list. How to insert an item in the linked list after a given node.**
  - **Code snippet for performing a recursion with a given problem**
  - **Write a recursive function to reverse a string,**
  - **Write a function to reverse a linked list**
  - **Write a function to insert an item in stack**
  - **A code snippet to dynamically allocate memory and perform some operation and free the memory**

Question pattern continue…

- **Output tracing:** A code snippet will be provided and you will have to write the output.
- **Problem finding:** looking at a code you should say whether a memory leak is happening or a segmentation fault is occurring in the code!
- **Explaining a code:** Describing a give code
- Showing the steps of Binary search and writing related codes.
- **Complexity analysis**
  - Basics of run-time analysis
  - Deriving run-time from a piece of code.
  - Math related to execution time calculation from Big-O
  - Deriving run-time of a given code snippet/function
  - Solving summation
  - Code to summation and then run-time analysis
  - Showing the steps of calculating run time of binary search
  - Solving recurrence relation, result of different iteration and substitution while solving.

How to prepare

- Remember, this is **NOT** a course where I will say to memorize the factorial function for recursion, memorize the push and pop function for stack, memorize the insert beginning function of linked list.
  - But we saw, how a factorial is calculated and how we can design a recursive function to calculate that.
  - With examples, we have seen how recursion put the information in a function call stack and seen recursion tree.
  - How to think about linked list node and how to join them, how to remove one node, how to traverse linked list to access the nodes to do various operations, etc.
- So, this course is more about understanding the concepts, apply those concepts in coding to similar or other unseen problems, etc.
- So, in the exam, you will demonstrate your understanding by answering the conceptual questions both in terms of design and coding.
- But, of-Couse you need to remember what a push and pop function does, how a linked list work, logic behind inserting an item at the end of a linked list, etc.
- Understanding how recursion works and draw the process.

How to prepare

- You have already seen the quiz questions, have seen the slide about the question pattern. So, review the course materials keeping the question pattern and past quizzes in mind.
- Go through the slides and examples, class notes that you have taken, and very importantly exercises (not memorizing. Understand them) and try them on your own without looking at the solutions.
- Uploaded codes that we have discussed are also very important!
- Go through the lab problems, too.
- Practice [If you have not done so yet, again try to do the exercises on your own, try to do the examples in the slides on your own without looking at the solution]
- Again, if there is something in the slide, try to do it yourself and then see the solution.
- Mapping concept to code is very important. Slides, exercises, and lab problems have enough such concepts to code mapping.

And at the end I must have to say:
I love you all and I wish you all the best.
Good luck for the exam!