Montana State University Computer Science Department

Graduating Senior Exam

Fall Semester 2021, 80 minutes.

Your performance helps us assess how well we are meeting our stated learning outcomes for computer science majors. To pass this exam, spend at least 45 minutes in order to provide a carefully reasoned answer to each question. This information will help us make improvements.

Name Jordan Stacy

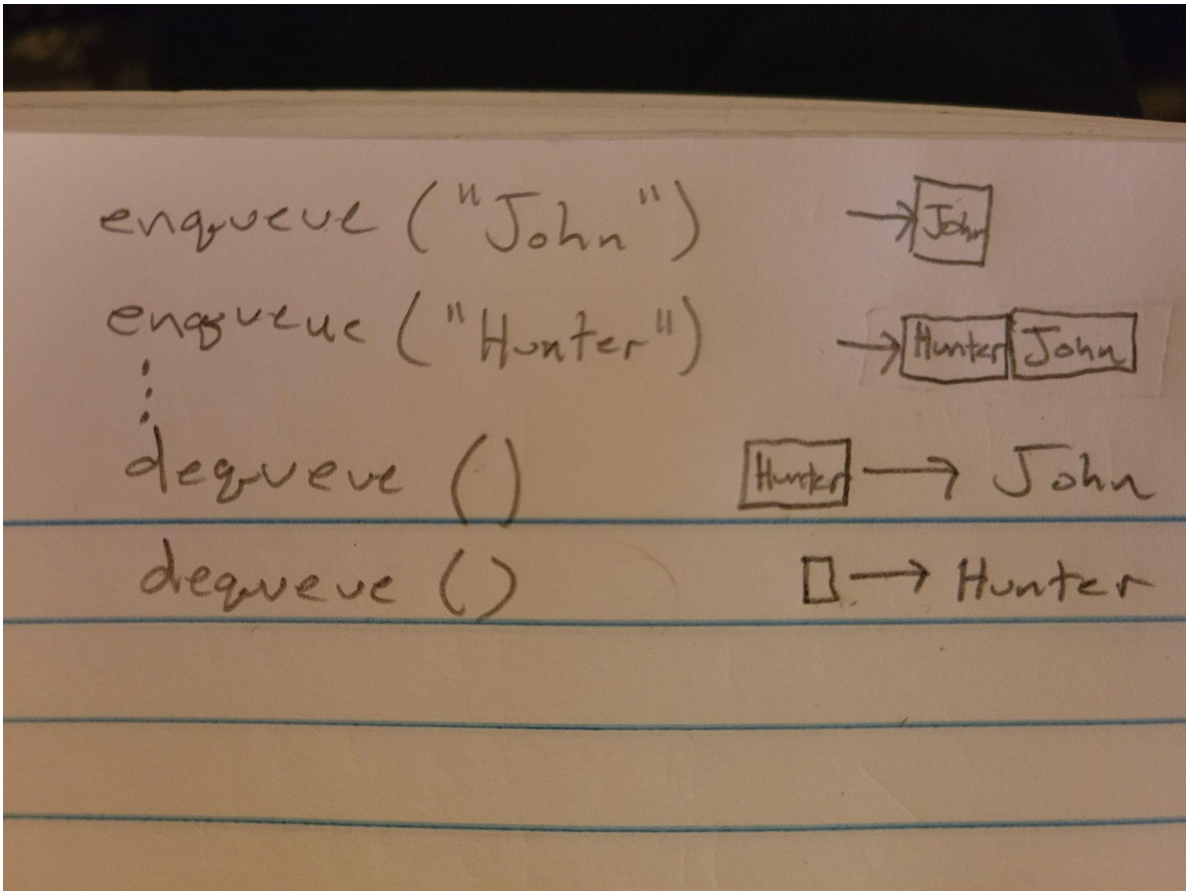
Question 1: A company uses a ticket system for customer service. Each ticket is given an urgency score when it is entered into the system. For example, a VIP customer would receive a higher urgency score than a regular customer. Assume that the urgency scores range from 1 to 10,000. The company's customer service team needs a way to find the most urgent ticket in the system and remove it when it has been addressed. New tickets can arrive at any time.

a) What simple data structure would best model this situation? Explain.

A queue would be the best due to the urgency score being assigned the moment the customer is entered into the system. The customer would be added, then enqueued onto the head of the queue. When the customer is processed, they are then dequeued from the queue. Queues are a first in first out data structure that is ideal for this kind of situation.

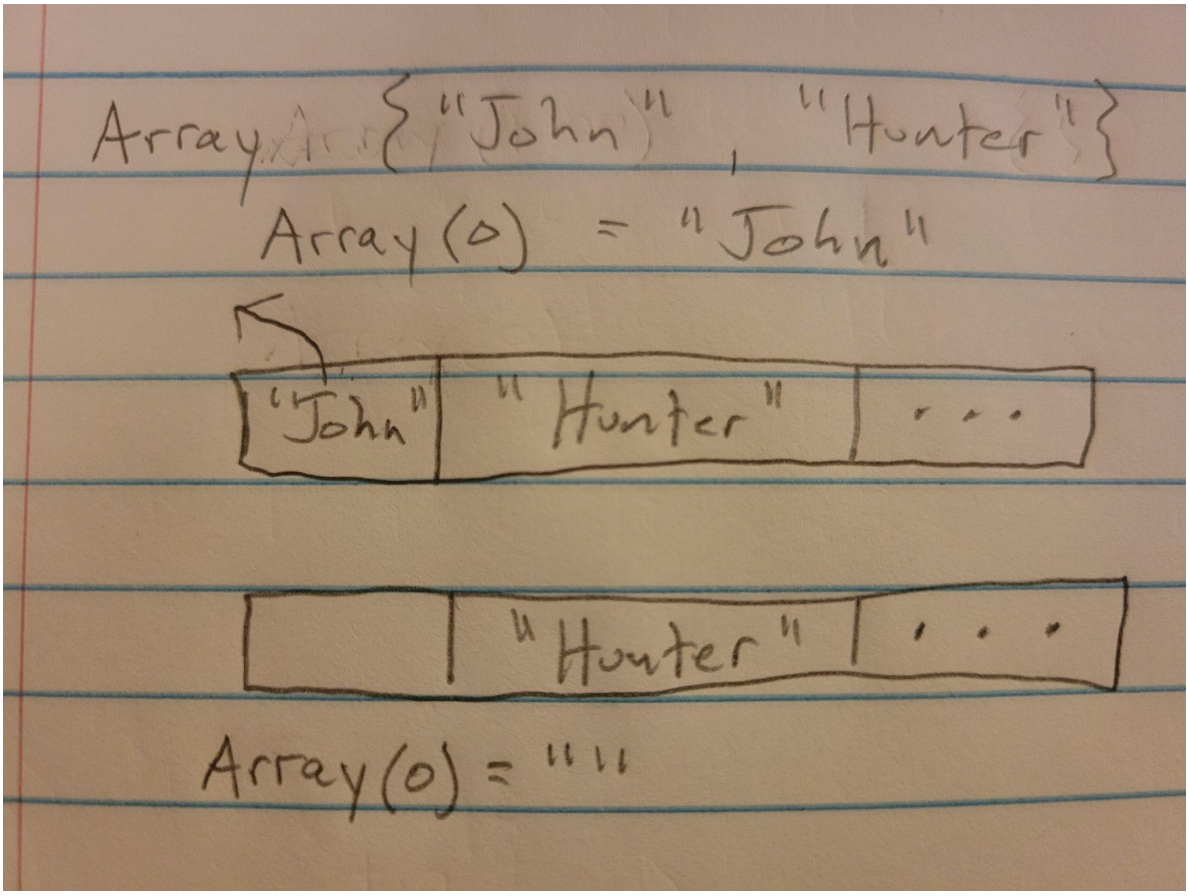
b) Describe one way in which this data structure could be implemented and draw a picture of that data structure.

As mentioned above, the ideal way in my opinion is using a queue. It can be implemented with the deque data structure as well, but a queue is sufficient. Each customer would be enqueued at the front of the queue. As they are helped, they would be dequeued, and the next customer would be placed at the head of the queue.



c) Describe a second way in which this data structure could be implemented and draw a picture of that data structure.

The second way could be the use of an array. It isn't ideal, but you can create an array of size 10,000 to hold all of the customers. Using a loop as they leave, you can "remove" them by replacing the value of each index to the empty string.



Question 2: Write a function in a language of your choice that has two parameters: a string and a character. The function should return the number of times that the character occurs in the string.

Your language of choice (e.g. Python): Java

Your solution:

```
public int numCharOccursInString(String str, char character){
        int count = 0;
        for (char letter : str.toCharArray()){
                if(letter == character){
                        count++;
                }
        }
        return count;
}
```

Question 3: Consider a singly linked list that contains **n** floating point numbers and is maintained in ascending order. An additional **k** floating point numbers will be added to the list, one at a time. You may assume that **k** << **n**.

Best case time complexity of algorithm: Omega(n)

Explanation of best case time complexity:

To search through the linked list, it would be Omega(n) time complexity. In order to insert the floating point values, it would be Omega(1). Linked Lists are not indexed, so the insertion would not affect the algorithm.

Worst case time complexity of algorithm: O(n)

Explanation of worst case time complexity:

Accessing the linked list, even with very large values would be O(n) complexity. Inserting values would be O(1). Since inserting values into a linked list is O(1), once inputs start getting larger, the computational weight lies on the accessing of the linked list. Therefore, the insertion will be removed from the complexity since the largest n value is what holds the most weight. Therefore the time complexity of the algorithm would be O(n).

Question 4: List any extracurricular community involvement (e.g. the Gallatin Valley Food Bank or coaching), extracurricular university involvement (e.g. Engineering Ambassadors, Swing Cats, AWC or ACM) or paid internships (e.g. Workiva intern or Undergraduate Course Assistant) that you were involved with during your time at MSU. For each activity, (1) **provide** the name of the organization, (2) **describe** briefly your main activities and (3) **estimate** the total number of hours you spent.

Organization	*Main Activities*	*Hours Spent (estimate)*
MSU Bozeman	*CSCI-127 Course Assistant*	*30*

Question 5: IEEE P7003 defines the Algorithmic Bias Standard.

a) Describe briefly what algorithmic bias is.

It is an unintended consequence of a created algorithm that usually benefits one or a few groups, but can set back certain groups of people based on certain social or economic reasons.

b) Provide a hypothetical example of how algorithmic bias could manifest itself in college admissions software.

An algorithm could be oversimplified by just admitting people in for certain placement test results. It is far more complicated than test scores. Some people of certain cultural backgrounds or where they were raised and went to school factors in on how they will place in an exam. Certain groups of people would be far less likely to be able to attend school if the algorithm was oversimplified by this. There must be other factors to weigh in before accepting or rejecting an individual.

Question 6: In 2010 Oracle sued Google claiming copyright and patent infringement for 11,500 lines of Java API copied code. Google used this small subset of code from the JAVA API for their mobile platform operating system. Pick a side (Google or Oracle) to win this lawsuit and explain 3 possible impacts to the computing industry of this win.

Side you choose to win: Google

First possible impact: If Google wins, this will help other projects build more software. In my personal opinion, APIs shouldn't be copyrighted in the first place. This will result in the stunting of progress by locking APIs for people or companies to use for their personal projects.

Second possible impact: If Oracle wins, this could put the free software movement at risk if all APIs became copyrighted. Free software usually fork other works and add on functionality. This would make doing these projects risky, and would discourage innovators from creating new things. Google winning could help keep the free software movement safe from these sorts of litigation.

Third possible impact: If Oracle won as well, this could escalate to algorithms to be copyrighted, and could reserve software to only the largest and most elite software groups who own the copyright. I may sound a bit like Richard Stallman here, but this could be a slippery slope to fall into. Google winning could help keep these policies from escalating.

Question 7: In the ACM Code of Ethics and Professional Conduct, one ethical principle stated is to ***avoid harm***. Describe three different types of harm that the software of an autonomous, self-driving car might cause.

First type of harm: Bodily harm. The camera sensors in the car could miss the detection of a child or animal in the street if it is dark, or they are hard to detect. This could lead to potential injury or death.

Second type of harm: Financial harm. This is partially linked to bodily harm, but the GPS system of the autonomous car could misread the location of another car, and smash another person driving a non self-driving car, or another self-driving car. This would bring up a dilemma of who would be sued and required to cover the costs of the accident. Would it be the self-driving car company, or the person operating the car?

Third type of harm: Emotional harm. If the software has any issues with operation, and the user cannot regain control of the car, this could result in potentially catastrophic harm to the users ability to trust the software in their vehicle.

Question 8: Describe your one-year plan following graduation. If you plan to work, list where you have applied (the company and the position) and mention whether you have received any offers. If you plan to go to graduate school, list where you have applied (the school and the area of study) and mention whether you have received any acceptances.

In one year, I plan to be working. I have applied to Wells Fargo. They have a Java Software Engineer position, but I haven't heard back yet. I plan to search out some more positions when the flow of the semester slows down a bit.
I am interested in a lot of different fields of computer science, and am flexible on what my end goal is. Due to the sheer amount of options in this field, I haven't found a total focus yet. I hope once I see all of the options available in the professional field, I can hone in a focus.

Question 9: List any organized social activities that you were involved with during your time at MSU. For each activity, (1) **provide** the name of the organization, (2) **estimate** the total number of hours you spent.

Organization Name	*Hours Spent (estimate)*
N/A	

Question 10: List any activities that you were involved with during your time at MSU that prepared you to work more effectively with teammates whose backgrounds are significantly different than your own. For each activity, (1) **describe** the activity and (2) **estimate** the total number of hours you spent.

Activity Description	*Hours Spent (estimate)*
CSCI-331 Final Team Project	*30 hours*
CSCI-221 Final Team Project	*15 hours*
CSCI-468 Senior Team Capstone	*40+ hours*

Question 11: Name the most sophisticated version control system that you use and describe three nontrivial features it has**.**

Name of most sophisticated version control system: Git

Non-trivial feature 1: It has the ability to collaborate and write code with others. It also gives others the ability, if the repository is public, to be able to look at the code and suggest edits, ask questions, or make a fork.

Non-trivial feature 2: You have the ability to edit code by pushing your code up to a remote server. This is very useful to be able to rollback to a previous version if the code somehow gets to an unworkable state.

Non-trivial feature 3: If any merge conflicts occur (two programmers code in the same place), you have the ability to compare both versions, and can either accept parts of each programmer's code, or fully accept one or the other.

Question 12: During your time at MSU, list any major non-classroom, non-paid activities that you engaged in to help prepare for your computer science career. For example, you might have taught yourself a programming language or attended a workshop on a CS topic. For each activity, **estimate** the number of hours spent**.**

Activity Description	*Hours Spent (estimate)*
Unity Programming (not Hunter's class)	*30 hours*
C++ Tutorials	*10 hours*
Android Studio Applications	*15 hours*

Best wishes wrapping up the semester and in the future!