Homework - Data Structures and Complexity

Due: 03-13-2020 (Friday @ 3:30 p.m.)

- Given a collection of algorithms that runs on **O(1), O(n log n), O(n), O(n^2), O(log n), O(n!)**, order the algorithms from fastest to slowest. **Answer: O(1),O(log n), O(n), O(n log n), O(n^2), O(n!)**
- Suppose that the complexity of an algorithm is O(n^2). Suppose that the program that uses the algorithm run in 10 seconds for a data set of size n. If the data size is doubled, how long will it take (approximately) to run the program? **Answer: 40 seconds, because n --> 10 , 2(n) --> , 20^2 = 40 **
- Complexities : **O(1), O(n log n), O(n), O(n^2), O(log n), O(n!) , O(h)**

1.Finding the max element in an unordered stack would require _____ operations? **Answer: O(n) because, finding the max element searches through all elements in the stack. **

2. Finding the min element in an unordered queue would require _____ operations? **Answer: O(n), because finding the min element searches through all elements in the queue. **
3. Finding some element in some Binary Search Tree would require _____ operations? **Answer: O(h) or O(n), because O(h) finds the element based on the height of the tree , O(n) when height of the tree becomes n.**
4. Finding some element in a balanced Binary Search Tree would require _____ operations? **Answer: O(log n), because binary search because the Binary Search Tree is balances.**
5. Finding some element in an ordered linked list would require _____ operations (worst case)? **Answer: O(n), because searches through all elements in the linked list.**
6. Finding some element in an ordered linked list would require _____ operations (average case)? **Answer: O(n/2), because searching through elements half of the linked list will be either less or more than some element we are finding on average.**
7. Finding some element in an unordered linked list would require _____ operations (worst case)? **Answer: O(n), because searches through all elements.**
8. For each of the following, count the number of operations where some_statement is executed based on the loops

```
//A
for (int I = 0; I < n; I++)
    for (int j = 1; j < n; j++)
        {some_statement;}
```

Answer: O(n^2), because "I" loops through n times and "j" loops through n times, which leads to n X n or n^2.

```
//B
for (int I = 0; I < n; I +=2)
```

```
    for (int j = 1; j < n; j++)
        {some_statement;}
```

Answer: O(n^2),Even though I iterates plus 2 each time "I" loops through n times and "j" loops through n times, which leads to n X n is n^2.

```
//C
for (int j = 1 ; j < n ; j *= 2)
    for (int I = 1; i<n; i++)
        {some_statement;}
```

Answer:O(n log n), because the first loop goes through log n times with "j" iterating to the power to 2. The second loop "I" goes through n times.

At most, how many comparisons are required to search a sorted vector of 1023 elements using the binary search algorithm?

Answer: At most 10 comparisons will be required to search a sorted vector of 1023 elements using the binary search element.

In each of the following examples, please choose the best data structure(s).

- Options are: **Array, Linked Lists, Stack, Queues, Trees, Graphs, Sets, Hash Tables**.
- Note that there may not be one clear answer.

1. You have to store social network "feeds". You do not know the size, and things may need to be dynamically added. **Answer: Linked Lists, because memory is dynamically allocated**
2. You need to store undo/redo operations in a word processor. **Answer: Stack, because stack is Last In First Out.It can pop off and pop on and store your undo/redo operations**
3. You need to evaluate an expression (i.e., parse). **Answer:Stack, because stack is LIFO, where the expression is evaluated from the top of the stack**
4. You need to store the friendship information on a social networking site. I.e., who is friends with who. **Answer:Graphs, because the nodes can relate the relationship between each person.**
5. You need to store an image (1000 by 1000 pixels) as a bitmap. **Answer:Array, because direct access to each element containing the pixel parts.**
6. To implement printer spooler so that jobs can be printed in the order of their arrival. **Answer:Queues, because queue is based on FIFO, meaning the first order will be the first print to arrive.**
7. To implement back functionality in the internet browser. **Answer:Linked List, because you can use the previous node to go back.**
8. To store the possible moves in a chess game. **Answer:Graphs, because you can use the nodes to store the relationship (what has been used and not) between each move made.**
9. To store a set of fixed key words which are referenced very frequently. **Answer:Hash Tables, because it is used to uniquely identify the key, stores the key and accesses it easily.**
10. To store the customer order information in a drive-in burger place. (Customers keep on coming and they have to get their correct food at the payment/food collection window.) **Answer:Queues, because queues are FIFO, where the first in the drive-in burger place will be the first to have there information stored.**
11. To store the genealogy information of biological species. **Answer:Trees, because the nodes can track the relationship and store the geneology information.**
12. To store an alphabetized list of names in order to look up quickly. **Answer:Trees, because a balanced tree where the nodes are arranged from A-Z can be accessed quickly.**

Deliverables

- Edit this file and add your answers using markdown!
- Create a folder called `H03` in your assignments folder.

- Put a copy of your markdown file in this folder, and call it README.md.
- Upload to github sometime close to the due date.
- Print out your banner ON ITS OWN PAGE

```
H03
3013
LASTNAME
```

- Print out a hard copy of the file as well. Do not print directly from github. Either use `gitprint` or make it a pdf and print it.
- Make sure you answer thoroughly using complexities where appropriate and/or explaining your choices etc.