United International University (UIU)

Dept. of Computer Science & Engineering (CSE)

Mid assessment Assignment 1: Summer 2024
Course Code: CSE 2215, Course Title: Data Structures and Algorithms 1
Total Marks: 10 Deadline: 18.09.24

---

1. Apply Quick Sort to the given array to sort it in Descending order. You must demonstrate each step of the simulation in detail. Choose the pivot strategy that is most likely to achieve the best-case scenario for the following array:

   Array: 20 33 41 55 56 60 61 70

2. Imagine you are a librarian who needs to organize a small stack of returned books on a shelf. The books are initially in a random order. You decide to use the Insertion Sort algorithm to sort the books by their titles.
   Given that you often add one book at a time to the already sorted section of the shelf, explain how the time complexity of Insertion Sort makes it a suitable choice for this task. Additionally, compare its performance in the best-case scenario versus the worst-case scenario when sorting the stack of books. Provide examples to illustrate your explanation.

3. In a magic land, numbers are sorted in a special order. You need to find the magic number 77 in the following sorted list of numbers: [1, 4, 7, 13, 26, 37, 50, 62, 77, 88, 99]. You can use either linear search or binary search. But beware, if you use more than 3 steps, the spell will break! Which search method will you choose, and can you find the number in 3 steps or less?

4. Find the memory location of A[15][45] if loc(A[5][15])=4500. Assume row-wise memory is allocated in the double type array A[60][60], where each double data is 8 bytes.

5. Consider the following two functions and determine the time complexity of both:

   i.

```
n ← length[A]
for j ← 1 to n - 1
    do smallest ← j
        for i ← j + 1 to n
            do if A[i] < A[smallest]
                    then smallest ← i
        exchange A[j] ↔ A[smallest]
```

ii.

```
int recFactorial( int n ){
   if( n == 1 )
     return 1;
   else
     return n * recFactorial( n-1 );
}
```

6. Prove the following:
   i. The running time of f(n) = $\frac{1}{3}n^3 - 5n^2$ is $\Omega(n^3)$.
   ii. The running time of f(n) = $5n^2 - 3$ is $O(n^2)$.
   iii. The running time of f(n) = $\frac{1}{2}n^2 - 7n$ is $\theta(n^2)$.

United International University (UIU)

Dept. of Computer Science & Engineering (CSE)

Mid assessment Assignment 2: Summer 2024
Course Code: CSE 2215, Course Title: Data Structures and Algorithms 1
Total Marks: 10 Deadline: 25.09.24

1.

Circular doubly linked lists are commonly used to implement music playlists, allowing users to easily navigate between tracks and repeat the playlist once the last track is reached. Suppose you wanted to add two pieces of data for each node : one is the artist's name and the other is the song name. Remember, the nodes should be **ordered alphabetically** (dictionary wise) by the artist's name so that the music player will play them in order.

i. Write the only portion code to define the node structure .
ii. Assume 3 node values for above and redraw the above figure with your values.

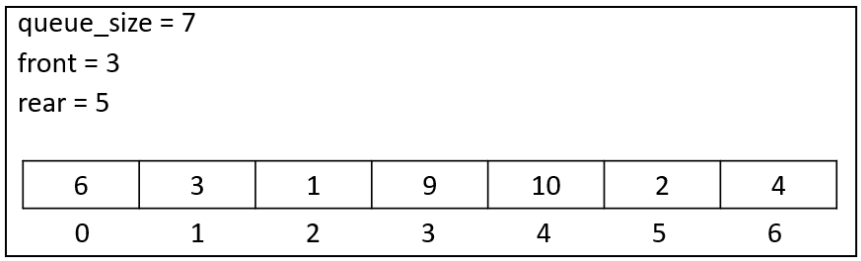
2. Draw and provide code to perform bellow operation on doubly linked lists:

i. Delete every node with a specific value from the list.
ii. Search for the middle element in the list and move it to the front of the list.

3. Write a function implementing a single linked list(data value integer) which can swap the first element and the last element of the list(consider the head node as the parameter).

4. Implement a function called emptyStack() that removes and prints all the elements from a stack until it is empty. Assume that the stack is implemented using an array and that the usual push(int val) and pop() functions are already defined. Note that the stack consists of positive integers, and the pop() function returns 0 when the stack is empty.

5. Consider the circular queue in the given figure implemented using an array and the three variables queue_size, front and rear.



Answer the following questions based on the above diagram.

i. How many elements are currently in the queue. What are those elements?

ii. Draw the diagram of the queue again after executing the following operations, along with the values of the variables queue_size, front, and rear :
enqueue(10), enqueue(20), enqueue(30), dequeue(), enqueue(10), enqueue(90), enqueue(60), enqueue(70)

iii. If you wanted to implement a queue using a linked list, how many pointers would be required and why?

iv. A linked list based stack/queue will never result in overflow - do you agree with the statement? Give reason in favor of your answer.

6. Suppose you want to delete the minimum number in a stack s1[5] of numbers. You have an auxiliary stack s2[5] for temporary storage. Write down the code segment *void deletemin(int s1[])* to solve this problem. You can use the push and pop operation. Assume there are no duplicate numbers in the stack.