Department of Electrical Engineering
Assignment
Date: 21/08/2020

Course Details

Course Title: Data Structure and Algorithm **Module:** _____
Instructor: _____ **Total Marks:** 30

Student Details

Name: _____ **Student ID:** _____

Note: Plagiarism of more than 20% will result in negative marking.
Similar answers of students will result in cancellation of the answer for all parties.

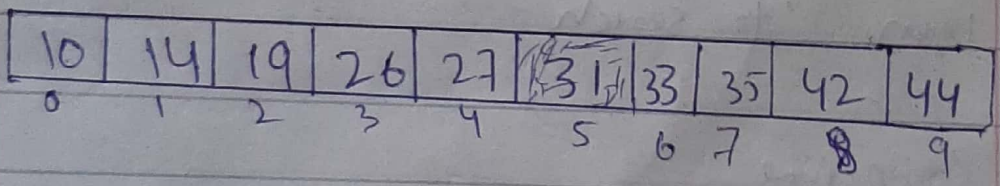
Q1.	The following is your sorted array and let assume that you need to search the location of value 31 using binary search. 	CLO 1 Marks 10
Q2.	Let LA be a Linear Array (Unordered) with N elements and K is a positive integer such that K<=N. Following is the algorithm where ITEM is inserted into the K^{th} position of LA- 1. Start 2. Set J=N 3. Set N= N+1 4. Repeat steps 5 and 6 while J>=K 5. Set LA [J+1]= LA[J] 6. Set J=J-1 7. Set LA[K]=ITEM 8. Stop Write the implementation of the above algorithm	CLO 2 Marks 10
Q3.	Find a given target number (x=61) using linear Search from a list of number using C++. [18, 36,56,61,73,87,93]	CLO 1 Marks 10

Name: Naveed Ali

ID No. 16753

Subject : Data structure and Algorithm

Q1. ~~Find~~ The following is your sorted array & let assume that you need to search the location of value 31 using binary search.



Ans

First of all we find the middle sorted Array using formula

$$Mid = Low + (high - Low)/2$$

Where Low = 0

high = 9

So,

$$Mid = 0 + \frac{(9-0)}{2} = 9/2 = 4.5$$

So middle term 4 whose value is 27.

10	14	19	26	27	31	33	35	42	44

Step 2.

Now comparing middle term to search

Comparing

Middle term < searching element

So half of array is left & search the right array.

Step 3

Low = mid + 1

$$Mid = Low + \frac{(high - Low)}{2}$$

So mid term = 7.
whose value is 35 which is greater than our target 31

10	14	19	26	27	31	33	35	42	44

10	14	19	26	27	31	33	35	42	44
					5				

So again Find Mid term which location 5 compare it with search value

both are same so our searching value 31 is in on location 5.

— X — X — X — X —

Q2. Let LA be a Linear Array (Unordered) with N element & K is positive integer such that K <= N following is the algorith where ITEM is inserted the K position of LA.

1. Start
2. Set ~~N=N+1~~ J=N
3. Set N= N+1
4. Repeat step 5 and 6 while J>=K
5. set LA[J+1] = LA[J]
6. Set J=J-1
7. set LA[K] = ITEM
8. Stop

Ans.

Implementation of the above algorithm

```
#include <stdio.h>

main [] {
 int LA [] = {1,3,5,7,8};

 int item =10, k = 3, n=5;
    int i = 0, J=n;
 printf ("The orisinal array element are
      :/n");
   for (i=0; i<n; i++) {

Printf ("LA[%d] = %d /n", i,LA[i]);
                                    }
```

```
n=n+1;
  while (j>=k){
  LA[J+1] = LA [J];
     j = j-1;
     }
  LA[k] = item
printf ["The array element after insertion
: /n");

for(i=0; i<n; i++){

printf ("LA[%d] = %d /n", i, LA[i]);
 }
 }
```

When compile & execute, above program produce the following result

The original array ~~elevat~~ elements are:

LA[0]=1

LA[1]=3

LA[2]=5

LA[3]=7

LA[4]=8

The array element for inseration:

LA [0] = 1

LA [1] = 3

LA [2] = 5

LA [3] = 10

LA [4] = 7

LA [5] = 8

— x — x — x — x — x —

Q3. Find a given target number (x = 61) using Liner search from list of number using C++

[18, 36, 56, 61, 73, 87, 93].

Ans:

```
#include <iostream>
using namespace std;
  int main () {
    int seq , x, n=7.
  int arr [] = {18.36, 56, 61, 73, 87,93};
    seq = 61;
```

```
for (x=0; x<n; x++){
    if (arr[x]==sea){
printf ("%d is present at location
  %d. /n", search, x+1);
    break;
  }
}
  if (x==n)
printf ("%d isn't present in the
array. /n", search);

  return 0;
        }
```

7.

—— x —— x —— x —— x ——