

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

Course Details

| | |
|--|-------------------------------|
| Course Title: <u>Data Structure and Algorithm</u> | Module: _____ |
| Instructor: _____ | Total Marks: <u>30</u> |

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. | <p>The following is your sorted array and let assume that you need to search the location of value 31 using binary search.</p> <div style="text-align: center;"><table border="1" style="display: inline-table; border-collapse: collapse;"><tr><td style="padding: 5px;">10</td><td style="padding: 5px;">14</td><td style="padding: 5px;">19</td><td style="padding: 5px;">26</td><td style="padding: 5px;">27</td><td style="padding: 5px; background-color: #e0f0ff;">31</td><td style="padding: 5px;">33</td><td style="padding: 5px;">35</td><td style="padding: 5px;">42</td><td style="padding: 5px;">44</td></tr><tr><td style="text-align: center;">0</td><td style="text-align: center;">1</td><td style="text-align: center;">2</td><td style="text-align: center;">3</td><td style="text-align: center;">4</td><td style="text-align: center;">5</td><td style="text-align: center;">6</td><td style="text-align: center;">7</td><td style="text-align: center;">8</td><td style="text-align: center;">9</td></tr></table></div> | 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | CLO 1 Marks 10 |
| 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 | | | | | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | | | | | | | | | | | | |
| Q2. | <p>Let LA be a Linear Array (Unordered) with N elements and K is a positive integer such that $K \leq N$. Following is the algorithm where ITEM is inserted into the K^{th} position of LA-</p> <ol style="list-style-type: none">1. Start2. Set $J=N$3. Set $N= N+1$4. Repeat steps 5 and 6 while $J \geq K$5. Set $LA [J+1]= LA[J]$6. Set $J=J-1$7. Set $LA[K]=ITEM$8. Stop <p>Write the implementation of the above algorithm</p> | CLO 2 Marks 10 | | | | | | | | | | | | | | | | | | | | |
| Q3. | <p>Find a given target number ($x=61$) using linear Search from a list of number using C++.</p> <p>[18, 36,56,61,73,87,93]</p> | CLO 1 Marks 10 | | | | | | | | | | | | | | | | | | | | |

①

Name: Hazrat Bilal

ID: 14622

Course Title: Data structure & Algorithms

Date: 21/08/2020

Marks: 30

Department: Electrical engineering.

Q No1:- The following is your sorted array and let assume that you need to search the location of value 31 using binary search.

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Ans:- Find location of 31 by binary search.

The follow is our sorted Array.

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

2

First we shall determine
half of the array by
using this formula.

$$\text{mid} = \text{low} + (\text{high} - \text{low}) / 2$$

Here it is
 $0 + (9 - 0) / 2 = 4$

So 4 is the mid of the
array.

↓

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Now we compare the value stored
at location 4 with the value
being searched i.e. 31. we find
that the value at location
4 is 27, which is not
match. As the value is greater
than 27 and we have to
sorted array, so we know
that the target value must be
in the upper portion of the array.

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

we change our low to mid+1 and
find the new mid value again.

(3)

$$\text{low} = \text{mid} + 1$$

$$\text{mid} = \text{low} + (\text{high} - \text{low}) / 2$$

our new mid is 7 now. we compare the value stored at location 7 with our target value 31.

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

The value stored in location 7 is not match, rather it is more than ~~that~~ what we are looking for. So, the value must be in the lower part from this location.

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Hence we calculate the again this is 5.

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

we compare the value stored at location 5 with our target value

(4)

we find that it is match.

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 14 | 19 | 26 | 27 | 31 | 33 | 35 | 42 | 44 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

we conclude that the target value 31 is stored at location 5.

ii ii ii ii ii ii

Q No 2:-

Let LA be a linear Array (unordered) with N elements and K is a positive integer such that $K \leq N$. Follow is the algorithm where ITEM is inserted into the Kth position of LA.

- 1) Start
- 2) set $J = N$
- 3) set $N = N + 1$
- 4) Repeat steps 5 and 6 while $J \neq K$
- 5) set $LA[J+1] = LA[J]$
- 6) set $J = J - 1$
- 7) set $LA[K] = \text{ITEM}$
- 8) Stop.

Write the implementation of the above algorithm.

5

Ans: Below is the 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 original array elements 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 elements after insertion
```

```
for (i = 0; i < n; i++) { : \n");
```

⑥

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

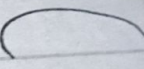
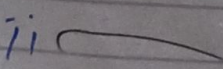


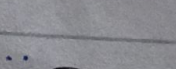
when compile and execute, above program produces the following result.

The original array elements are:

LA[0] = 1
LA[1] = 3
LA[2] = 5
LA[3] = 7
LA[4] = 8

The array elements after insertion:

LA[0] = 1
LA[1] = 3
LA[2] = 5
LA[3] = 10
LA[4] = 7
LA[5] = 8.

ii  ii  ii  ii  ii 

Q NO 3:- Find a given target number (x=61) using linear search from a list of number.

⑦

using C++ [18, 36, 56, 61, 73, 87, 93]

Ans:-

```
#include <iostream>
using namespace std;
int search(int arr[], int n, int x)
{
    int i;
    for(i=0; i<n; i++)
        if(arr[i] == x)
            return i;
    return -1;
}
int main()
{
    int arr[] = {18, 36, 56, 61, 73, 87, 93};
    int n = sizeof arr /
              sizeof(arr[0]);
    int x = 61
```


8

```
int index = search(arr, n, x);
```

```
if (index == -1)
```

```
    cout << "Element is not present  
    in the array";
```

```
else
```

```
    cout << "Element found at  
    position" << index;
```

```
    return 0;
```

```
}
```

output

The element found
at index 3.

ii ~ ii ~ ii ~ ii ~ ii