NAME : Badrul Zaman

ID : 13685

paper: Data Structure

Q1 :=>

Answer :=>

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

first we shall determine half of the array by using this formula-

mid = low + (high - how)/2

Here it is 0+(9-0)/2 = 4 (integer value of 4.5) so

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 fenel that the value at location 4 is 27. which is not match As the value greater then 27 and we have sorted array so we also know that the target value must be in the upper portion of the array.

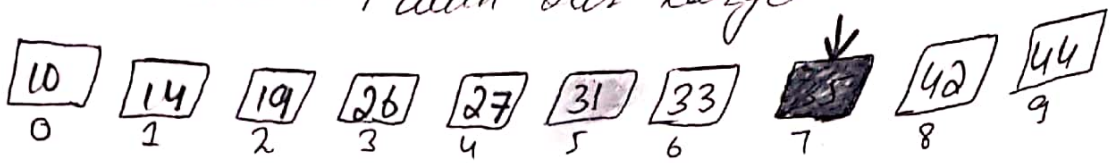
10	14	19	26	27	31	33	35	42	44
0	1	2	3	4	5	6	7	8	9

we changed our low to mid + 1 and fenel the new mid value again

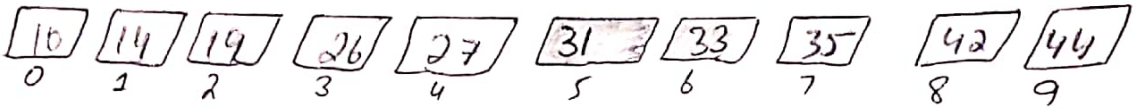
low = mid + 1

mid = low + (high - low)/2.

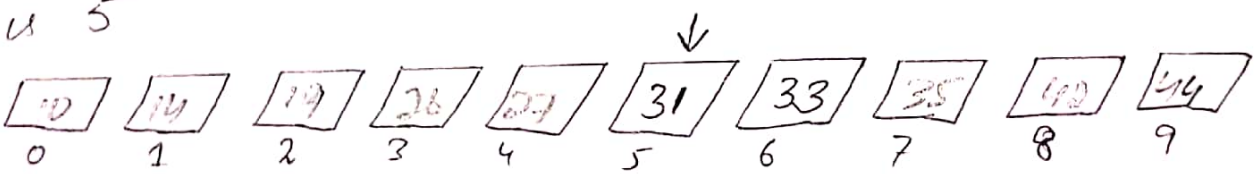
Our new mid is 7 Now. we compare the value stored at location 7 with our target value.



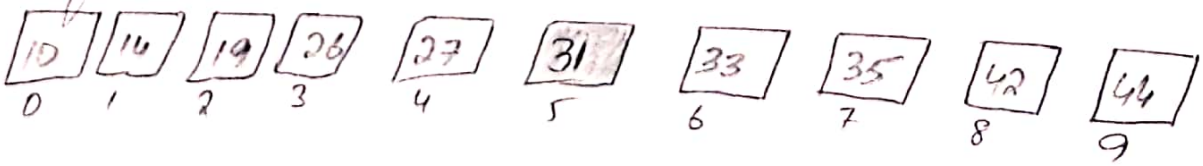
The value store at location 7 is not a match rather it is more then what we are looking for. So. the value must be in the lower part from this location



Hence, we calculate the mid again This time it is 5



We compare the stored at location 5 with our target value we find it is a match



We conclude that the target value 31 is stored at location 5. Binary search halves the searchable items and thus reduces the count of comparisons to be made to very less numbers.

Q2 :→

Ans :→

```
#include <stdio.h>
 main(){
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]); } h = n+1;
while (j>=k)
{
LA[J+1] = LA[j];
  j = j-1;
}
LA[K] = item;
printf(The array elements after insertion:\n");
for(i=0; i<n; i++){
printf("LA[%d] = %d \n", i, LA[j]);
```

```
}
}
```


Output:

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

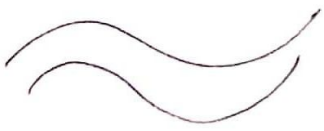
LA[1] = 3

LA[2] = 5

LA[3] = 10

LA[4] = 7

LA[5] = 8



Q5:-

(5)

```
#include <iostream>
using namespace std;
void linearSearch(int a[], int n)
    int Temp = -1;
for (int i=0; i<7; i++)
{
 if (a[i] == n)
 {
 cout << "Element found at location: " << i << endl;
 }

}

}
int main()
 int arr[7] = [18, 3656, 61, 73, 87, 93]
cout << "please Enter an elements to Search " << endl;
 int num;
 Cin >> num;
linearSearch(arr, num);
   return 0;
}
```