Linear search

Implementation:

```
#include<bits/stdc++.h>
using namespace std;
int n;
int  linearSearch(int ara[],int x)
{
    for(int i=0; i<n; i++)
        if(x==ara[i])
            return i;
    return -1;
}
int main()
{
    int ara[]= {2,1,4,108,98,7},x,res;
    n=sizeof(ara)/sizeof(ara[0]);
    cout<<"Enter element you want to search(press 0 to exit) : ";
    while(1)
    {
        cin>>x;
        if(x==0)
            break;
        res=linearSearch(ara,x);
        if(res==-1)
            cout<<x<<" is not found\n";
        else
            cout<<x<<" is found at position "<<res+1<<endl;
    }
    return 0;
}
```

S.M. Khasrul Alam Shakil
191-15-12180

Analysis:

```
int  linearSearch(int ara[],int x)
{
   for(int i=0; i<n; i++)
      if(x==ara[i])
         return i;
   return -1;
}
```

Suppose have a Array of 5 element,

Ara={10,15,35,5,25}

Let x=20, we want know the existence of x in the array so the procedure is given below:

<u>Step 1:</u>

Compare x with 1^{st} element of the array

10	15	35	5	25

ara[0], x!=ara[0] ,go to the next step

<u>Step 2:</u>

10	15	35	5	25

Compare x with 2^{nd} element of the array

ara[1], x!=ara[1] , go to the next step

<u>Step 3:</u>

10	15	35	5	25

Compare x with 3^{rd} element of the array

ara[1], x!=ara[2] , go to the next step

<u>Step 4:</u>

10	15	35	5	25

Compare x with 4^{th} element of the array

ara[1], x!=ara[3] , go to the next step

<u>Step 5:</u>

10	15	35	5	25

Compare x with 5^{th} element of the array

X==ara[4] ,so function will return index of ara[4]

S.M. Khasrul Alam Shakil
191-15-12180

Time complexity

Best case:

We must know the case that causes minimum number of operations to be executed. In the linear search problem, the best case occurs when x is present at the first location. The number of operations in the best case is constant (not dependent on n). So time complexity in the best case would be: Θ(1)

Worst case:

For Linear Search, the worst case happens when the element to be searched (x in the above code) is not present in the array. When x is not present, the search() functions compares it with all the elements of arr[] one by one. Therefore, the worst case time complexity of linear search would be : Θ(n).

Average case:

For average case Sum all the calculated values and divide the sum by total number of inputs.

$$= \frac{1+2+3\ldots\ldots+n}{n}$$

$$= \frac{\frac{n(n+1)}{2}}{n}$$

$$= \frac{n+1}{2}$$

Ignoring the constant co-efficient, we can say that the complexity in average case of linear search is : O(n).

S.M. Khasrul Alam Shakil
191-15-12180