

Assignment No. 4A

Write a python program to store roll numbers of student in array who attended training program in random order. Write function for searching whether particular student attended training program or not, using Linear search and Sentinel search.

PROBLEM STATEMENT:

A) WRITE A PYTHON PROGRAM TO STORE ROLL NUMBERS OF STUDENT IN ARRAY WHO ATTENDED TRAINING PROGRAM IN RANDOM ORDER. WRITE FUNCTION FOR SEARCHING WHETHER PARTICULAR STUDENT ATTENDED TRAINING PROGRAM OR NOT, USING LINEAR SEARCH AND SENTINEL SEARCH.

Theory:

Linear Search :

The idea behind linear search is to compare the search item with the elements in the list one by one (using a loop) and stop as soon as we get the first copy of the search element in the list. Now considering the worst case in which the search element does not exist in the list of size N then the Simple Linear Search will take a total of $2N+1$ comparisons (N comparisons against every element in the search list and N+1 comparisons to test against the end of the loop condition).

Sentinel Linear Search :

Here the idea is to reduce the number of comparisons required to find an element in a list. Here we replace the last element of the list with the search element itself and run a **while loop** to see if there exists any copy of the search element in the list and quit the loop as soon as we find the search element. See the code snippet for clarification.

Algorithm:

```
int last = array[N-1];

array[N-1] = item;    // Here item is the search element.
int i = 0;
while(array[i] != item)
{
    i++;
}
array[N-1] = last;
if( (i < N-1) || (item == array[N-1]) )
{
    cout << " Item Found @ "<<i;
}
else
{
    cout << " Item Not Found";
}
```

```
}
```

ANALYSIS :

Here we see that the **while loop** makes only one comparison in each iteration and it is sure that it will terminate since the last element of the list is the search element itself. So in the worst case (if the search element does not exists in the list) then there will be at most $N+2$ comparisons (N comparisons in the while loop and 2 comparisons in the if condition). Which is better than ($2N+1$) comparisons as found in **Simple Linear Search**.

Take note that both the algorithms have time complexity of $O(n)$.

Program :

```
3
4 print("Sentinel Search.....")
5 rn=[1,2,3,4,5,44,55,30]
6 print(rn)
7 len=len(rn) # To find length of rn list
8 print(len) # To display len
9
10 key=30 # User wants to check either roll no 30 attended the training program or not
11 last=rn[len-1] # To store last element in last variable because we want to store
12 rn[len-1]=key # the key=30 value at the end of array or rn as a logic of sentinel search
13 print(last) # To print the last variable value
14 print(rn) # After replacing the last index value list rn element
15
16 i=0 # init of i = 0
17 while(rn[i]!=key): # while condition is true go on searching next element in rn once false come out
18     i=i+1
19
20 rn[len-1]=last # Store the last variable value to its original position so that it also get checked
21 if((i<len-1) or (key==rn[len-1])): # Check i is in scope of list or last element is equals to key.
22     print("key found at location:",i)
23 else:
24     print("Key not found....")
25
```

Output:

Sentinel Search.....

[1, 2, 3, 4, 5, 44, 55, 30]

8

30

[1, 2, 3, 4, 5, 44, 55, 30]

key found at location: 7