

Sub.: \_\_\_\_\_

SAT ☐ SUN ☐ MON ☐ TUE ☐ WED ☐ THU ☐ FRI ☐

DATE: / /

## Linear search

### Implementation:

```
int linear_search (int arr[], int n, int a)
{
    int i, index = -1 ;
    for (i = 0; i < n; i++)
    {
        if (arr[i] == a)
        {
            index = i;
            break;
        }
    }
    return index ;
}
```

### Analysis:

5	4	3	2	1
---	---	---	---	---

This is an array with 5 elements that

Sub.: \_\_\_\_\_

SAT ☐ SUN ☐ MON ☐ TUE ☐ WED ☐ THU ☐ FRI ☐

DATE: / /

means  $n=5$ . Now we want to search the value  $a=1$ . Firstly, we can see that,  $i=0$  of the alliteration all iteration which is less than  $n$ . So, it will enter in the loop. According to the condition, if the value is found then it will break and return the index. But in this case the value ~~we want~~ we want to search is in the last position or last index of array which is 4. So, the loop will continue for 5 times or  $n$  times here.

Time complexity (Linear Search)

Best case:

If  $a=5$  we can see, 5 is in the beginning of the array, For this the loop will run for only 1 time. So, time complexity would be  $O(1)$

Sub: \_\_\_\_\_

SAT ☐ SUN ☐ MON ☐ TUE ☐ WED ☐ THU ☐ FRI ☐

DATE: / /

### Average case:

we know average case = All possible case times  
Number of cases

$$= (1 + 2 + 3 + \dots + n) / n$$

$$= (n(n+1)/2) / n$$

$$= (n+1)/2$$

Ignoring the constant co-efficient, the  
time complexity in average case is  $O(n)$

### Worst case:

If there are  $n$  elements and the  
value either exists in the last position  
 $n-1$  or not exists, the loop will run  
for  $n$  times. So, the complexity in  
worst case of linear search is  $O(n)$