Class-1

Time Complexity
O(n)

memory Complexity
O(1)

```
a = [3,4,8,10,11,13]

target =8

for element in a:
    if target == element
        print ("found")
```

```
a = [1,8,13,5,4]
start =0
end = len(a)

while start <end:
    if a [start] = target
        print ("found")
    start+=1
```

start	end	t	a (start)	
0	5	5		
0	5	5	1	a[0]
1			8	a[1]
2			13	a[2]
3				a[3]
				a[4]

Scanned with CamScanner

Reversing an array (Using an empty array)

```
a = [1, 2, 3, 4]

result = [ ]

start = len(a) - 1

end = 0
while start >= 0 :

        result.append(a[start])
        start -= 1
print(result)
```

Result	Start
[ ]	3
[4]	2
[4, 3]	1
[4, 3, 2]	0
[4, 3, 2, 1]	-1

TC → O(n)
MC → O(n)

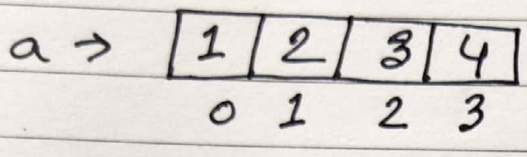
Scanned with CamScanner

Reducing time complexity by using a different algorithm (Two-pointer)

$a = [1, 2, 3, 4]$

T: $O(n)$

S: $O(n) \rightarrow O(1)$




```
s = 0          ← O(1)
e = len(a) - 1

while s < e:                    (n/2)
    temp = a[s]
    a[s] = temp
    s += 1
    e -= 1

return n
```

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

$= O(n/2)$

$= O(n)$

Scanned with CamScanner