# 6. INSERT ELEMENT AT POSITION IN AN ARRAY

**APPROACH:**

**1st** $We\ increment\ the\ size\ of\ the\ array$:

$$size = size + 1;$$

**2nd** $we\ shift\ the\ element\ from\ position\ to\ last\ index$:

$suppose\ size = 5\ and\ we\ have\ indexes$:

$a[0], a[1], a[2], a[3]\ and\ a[4]$.

$as\ size\ increased\ , now\ we\ have\ array\ indexes$:

$a[0], a[1], a[2], a[3], a[4]\ , a[5]\ and\ size = 6$.

$and\ we\ will\ put\ the\ at\ index\ 1$.

$$a[5] = a[4]\ [values\ get\ swapped]$$

$$a[4] = a[3][values\ get\ swapped]$$

$$a[3] = a[2][values\ get\ swapped]$$

$$a[2] = a[1][values\ get\ swapped]$$

$Now\ we\ just\ will\ do\ is\ overriding\ the\ i^{th}\ element$:

$$a[1] = elem\ (User\ Input)$$

**PROGRAM:**

```
 //Increment the array
        size=size+1;

        // Insert Elements in the array at a given
position
        for (int i = size-1 ; i > pos; i--)
        {

            a[i] = a[i - 1];

        }
        a[pos] = elem;
```

## TIME COMPLEXITY

***Therefore we see whole insertion and overriding of ith element takes:***

$$O(1)[Increment] + O(n)[Shift] + O(1)[Override] = O(n).$$