Parangat Bindal
IT
11912084.

5. Analysis of Insertion Sort

```
for (int x=1 ; x<n ; x++)
    {
        temp = arr[x]
    for (int y=x-1; y>=0 ; y--)
        { if (temp < arr[y])
            {
              temp [y+1] = temp[y] ;
              temp [y] = temp
            }
        else
        break;
    }
}
```

for best Case list should be in ascending order as we know Algorithm of insertion Sort.

Consider list is {7, 9, 11, 13, 16}

loop 1 - temp = 9.

loop 2 → 9 (9 < 7) else
false. break;

Means at x = 1, loop 2 has been called 1 time

Similarly.

(ii) for x = 2

temp = 11

loop 2 : x = 1 → y > 0 true

if (11 < arr[1]) then break

false.

Means at x = 2 loop 2 has been called once again

So we can say [In ascending order]

loop 1	loop 2	No. of iteration
x = 1	y = 0	1
x = 2	y = 1	1
x = 3	y = 2	1
x = 4	y = 3	1
x = 5	y = 4	1
⋮	⋮	
x = n-1	y = n-2	1

Total ~~for~~ loop calls n-1

Time complexity = O(n-1) = O(n) Ans