LAB#26

Example#1: Write a program to sort an unsorted array by using counting sort algorithm:

Solution:

```python
def counting_sort(initial_arr):
    max_val = max(initial_arr)

    count_arr = [0] * (max_val + 1)

    for num in initial_arr:
        count_arr[num] += 1

    for i in range(1, len(count_arr)):
        count_arr[i] += count_arr[i - 1]

    output_arr = [0] * len(initial_arr)

    for num in reversed(initial_arr):
        output_arr[count_arr[num] - 1] = num
        count_arr[num] -= 1

    return output_arr

initial_arr = [2,3,1,2,5,1,2,5,2,5]
sorted_arr = counting_sort(initial_arr)
print("Sorted array:", sorted_arr)
```

Result:

```
Sorted array: [1, 1, 2, 2, 2, 2, 3, 5, 5, 5]
```

Explanation:

Initial_arr=[2,3,1,2,5,1,2,5,2,5]

Sorted_arr=counting_sort(initial_arr)

def counting_sort([2,3,1,2,5,1,2,5,2,5]):

max_val=max(initial_arr)

max_val=5

count_arr=[0]*(max_val+1)

count_arr=[0]*6

count_arr=[0,0,0,0,0,0,0,0,0,0]

Count_arr→

0	1	2	3	4	5
0	0	0	0	0	0
	1	1	1		
	2	2			1
		3			2
0	2	4	1	0	3

for num in initial_arr:

Initial_arr=[2,3,1,2,5,1,2,5,2,5]

Count_arr[num]=Count_arr[num]+1

Count_arr[2]=Count_arr[2]+1=0+1=1

Count_arr[3]=Count_arr[3]+1=0+1=1

Count_arr[1]=Count_arr[1]+1=0+1=1

Count_arr[2]=Count_arr[2]+1=1+1=2

Count_arr[5]=Count_arr[5]+1=0+1=1

Count_arr[1]=Count_arr[1]+1=1+1=2

Count_arr[2]=Count_arr[2]+1=2+1=3

Count_arr[5]=Count_arr[5]+1=1+1=2

Count_arr[2]=Count_arr[2]+1=3+1=4

Count_arr[5]=Count_arr[5]+1=2+1=3

Count_arr→

0	1	2	3	4	5
0	2	4	1	0	3

Now, store the result of cumulative sum in the Count_arr:

Count_arr →

0	1	2	3	4	5
0	2	4	1	0	3
0	2	6	7	7	10

for i in range(1,len(Count_arr)):

Count_arr[i]=Count_arr[i]+Count_arr[i-1]

#i=1 Count_arr[1]=Count_arr[1]+Count_arr[1-1]

= Count_arr[1]+Count_arr[0]

=2+0=2

#i=2 Count_arr[2]=Count_arr[2]+Count_arr[2-1]

= Count_arr[2]+Count_arr[1]

=4+2=6

#i=3 Count_arr[3]=Count_arr[3]+Count_arr[3-1]

= Count_arr[3]+Count_arr[2]

=1+6=7

#i=4 Count_arr[4]=Count_arr[4]+Count_arr[4-1]

= Count_arr[4]+Count_arr[3]

=0+7=7

#i=5 Count_arr[5]=Count_arr[5]+Count_arr[5-1]

= Count_arr[5]+Count_arr[4]

=3+7=10

Updated Count_arr:

0	1	2	3	4	5
0	2	6	7	7	10

Output=[0]*len(initial_arr)

=[0]*10

Output=[0,0,0,0,0,0,0,0,0,0]

Now, we have:

Initial_arr →

0	1	2	3	4	5	6	7	8	9
2	3	1	2	5	1	2	5	2	5

Count_arr→

0	1	2	3	4	5
0	2	6	7	7	10

Output_arr→

0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0

for num in reversed(initial_arr):

#num=5

Output_arr[Count_arr[5]-1]=5

Output_arr[10-1]=5

Output_arr[9]=5

Count_arr[5]=Count_arr[5]-1

=10-1

Count_arr[5]=9

#num=2

Output_arr[Count_arr[2]-1]=2

Output_arr[6-1]=2

Output_arr[5]=2

Count_arr[2]=Count_arr[2]-1

=6-1

Count_arr[2]=5

#num=5

Output_arr[Count_arr[5]-1]=5

Output_arr[9-1]=5

Output_arr[8]=5

Count_arr[5]=Count_arr[5]-1

=9-1

Count_arr[5]=8

#num=2

Output_arr[Count_arr[2]-1]=2

Output_arr[5-1]=2

Output_arr[4]=2

Count_arr[2]=Count_arr[2]-1

=5-1

Count_arr[2]=4

#num=1

Output_arr[Count_arr[1]-1]=1

Output_arr[2-1]=1

Output_arr[1]=1

Count_arr[1]=Count_arr[1]-1

=2-1

Count_arr[1]=1

#num=5

Output_arr[Count_arr[5]-1]=5

Output_arr[8-1]=5

Output_arr[7]=5

Count_arr[5]=Count_arr[5]-1

=8-1

Count_arr[5]=7

#num=2

Output_arr[Count_arr[2]-1]=2

Output_arr[4-1]=2

Output_arr[3]=2

Count_arr[2]=Count_arr[2]-1

=4-1

Count_arr[2]=3

#num=1

Output_arr[Count_arr[1]-1]=1

Output_arr[1-1]=1

Output_arr[0]=1

Count_arr[1]=Count_arr[1]-1

=1-1

Count_arr[1]=0

#num=3

Output_arr[Count_arr[3]-1]=3

Output_arr[7-1]=3

Output_arr[6]=3

Count_arr[3]=Count_arr[3]-1

=7-1

Count_arr[3]=6

#num=2

Output_arr[Count_arr[2]-1]=2

Output_arr[3-1]=2

Output_arr[2]=2

Count_arr[2]=Count_arr[2]-1

=3-1

Count_arr[2]=2

Return Output_arr

[1,1,2,2,2,2,3,5,5,5]

Class Assignment

Q.1: Take an unsorted array, apply the counting sort algorithm to sort it, and then explain the code in detail, visualizing each step.