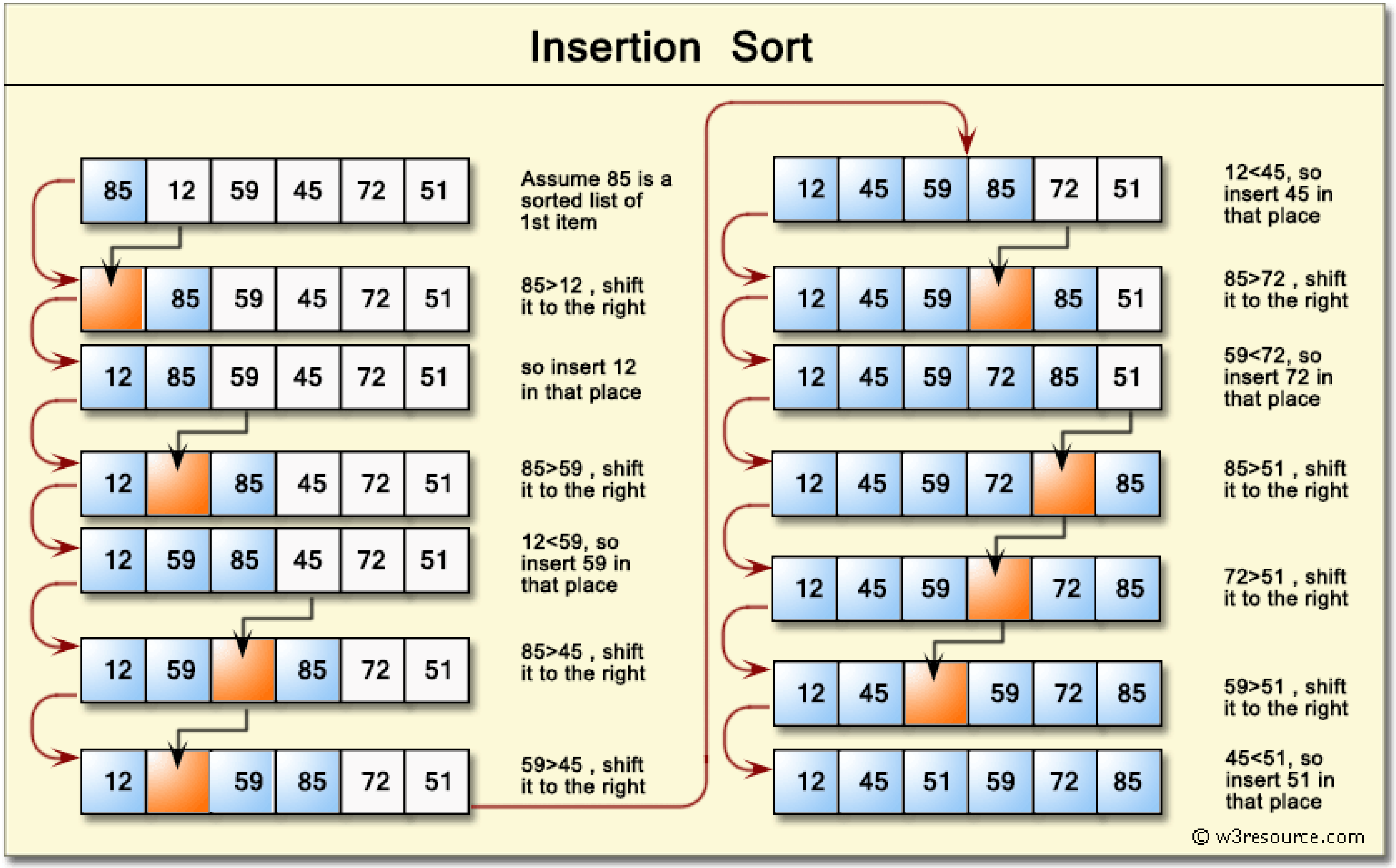
ऋते ज्ञानान्न मुक्तिः

GLA UNIVERSITY
MATHURA
Recognised by UGC Under Section 2(f)

Accredited with A Grade by NAAC

Insertion Sort

Insertion Sort



Insertion Sort

85	12	59	45	72	51	Assume 85 is a sorted list of 1st item
	85	59	45	72	51	85>12 , shift it to the right
12	85	59	45	72	51	so insert 12 in that place
12		85	45	72	51	85>59 , shift it to the right
12	59	85	45	72	51	12<59, so insert 59 in that place
12	59		85	72	51	85>45 , shift it to the right
12		59	85	72	51	59>45 , shift it to the right
12	45	59	85	72	51	12<45, so insert 45 in that place
12	45	59		85	51	85>72 , shift it to the right
12	45	59	72	85	51	59<72, so insert 72 in that place
12	45	59	72		85	85>51 , shift it to the right
12	45	59		72	85	72>51 , shift it to the right
12	45		59	72	85	59>51 , shift it to the right
12	45	51	59	72	85	45<51, so insert 51 in that place

© w3resource.com

Algorithm

```
// Sort an arr[] of size n
insertionSort(arr, n)
Loop from i = 1 to n-1.
……a) Pick element arr[i] and insert it into sorted sequence arr[0…i-1]
```

Algorithm

```
void insertion_Sort(arr[], n) //arr is an array, and n is size of array
{
    for i=1 to n do {
        key = arr[i];
        j = i - 1;
         while (j >= 0 && arr[j] > key) do
             {
              arr[j + 1] = arr[j];
              j = j - 1;
             }
         arr[j + 1] = key;
    }
 }
```

Algorithm

0	1	2	3	4	5	6	7
4	3	2	10	12	1	5	6

- N=8
- i=1,2,3,4,5,6,7
- Take i=1
- Key=a[i]=a[1]=3
- J=i-1=1-1=0
- Loop : j>=0 and a[j]>key i.e. a[0]>3 4>3 yes
- A[j+1]=a[j] a[1]=a[0]

4	4	2	10	12	1	5	6

- j=j-1=-1 loop will be false
- After loop a[-1+1]=3

3	4	2	10	12	1	5	6

Algorithm

0	1	2	3	4	5	6	7
3	4	2	10	12	1	5	6

- Take i=2
- Key=a[i]=a[2]=2
- J=i-1=2-1=1
- Loop : j>=0 and a[j]>key i.e. A[1]>2 4>2 yes
- A[j+1]=a[j] a[2]=a[1]
- 3 4 4 10 12 1 5 6
- j=j-1=1-1 =0   loop again run
- j>=0
- and a[j]>key i.e. A[0]>2 3>2 yes
- A[j+1]=a[j] a[1]=a[0]

3 3 4 10 12 1 5 6

- J=j-1 =0-1=-1 loop will stop
- A[j+1]=key

2 3 4 10 12 1 5 6

Algorithm

0	1	2	3	4	5	6	7
2	3	4	10	12	1	5	6

- Take i=3
- Key=a[i]=a[3]=10
- J=i-1=3-1=2
- j>=0 but a[2]>key 4>10 Loop will not execute
- A[3]=key
- A[3]=10 No change same

Algorithm

0	1	2	3	4	5	6	7
2	3	4	10	12	1	5	6

- Take i=4
- Key=a[i]=a[4]=12
- J=i-1=4-1=3
- j>=0 but a[3]>key 12>10 Loop will not execute
- A[3]=key
- A[4]=12 No change again same

Algorithm

0	1	2	3	4	5	6	7
2	3	4	10	12	1	5	6

Take i=5

Key=a[i]=a[5]=1

J=i-1=5-1=4

Loop j>=0 a[j]>key 12>1 Yes

A[j+1]=a[j] a[5]=a[4]

2	3	4	10	12	12	5	6

J=j-1=4-1=3

Loop j>=0 a[j]>key 10>1 Yes

A[4]=a[3]

2	3	4	10	10	12	5	6

Python Program

```
def insertionSort(arr):
    for i in range(1, len(arr)):
        key = arr[i]
        j = i-1
        while j >= 0 and key < arr[j] :
            arr[j + 1] = arr[j]
            j -= 1
        arr[j + 1] = key

arr = [12, 11, 13, 5, 6]
insertionSort(arr)
for i in range(len(arr)):
    print ("% d" % arr[i])
```

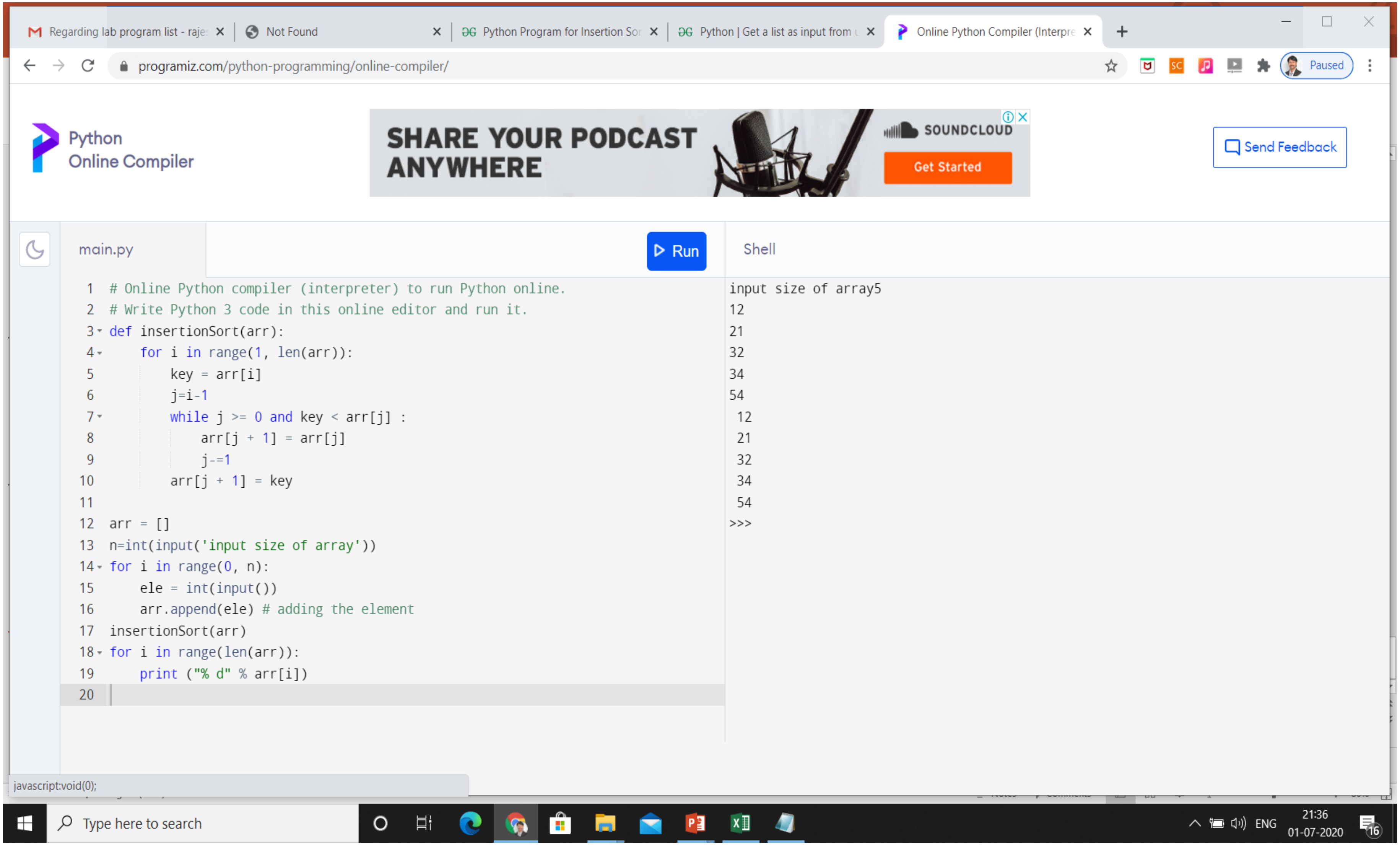
Python Program

```
def insertionSort(arr):
    for i in range(1, len(arr)):
        key = arr[i]
        j = i-1
        while j >= 0 and key < arr[j] :
                        arr[j + 1] = arr[j]
                        j -= 1
        arr[j + 1] = key
```

Python Program

```
arr = []
n=int(input('input size of array'))
for i in range(0, n):
        ele = int(input())
        arr.append(ele)
insertionSort(arr)
for i in range(len(arr)):
        print ("% d" % arr[i])
```

Execution

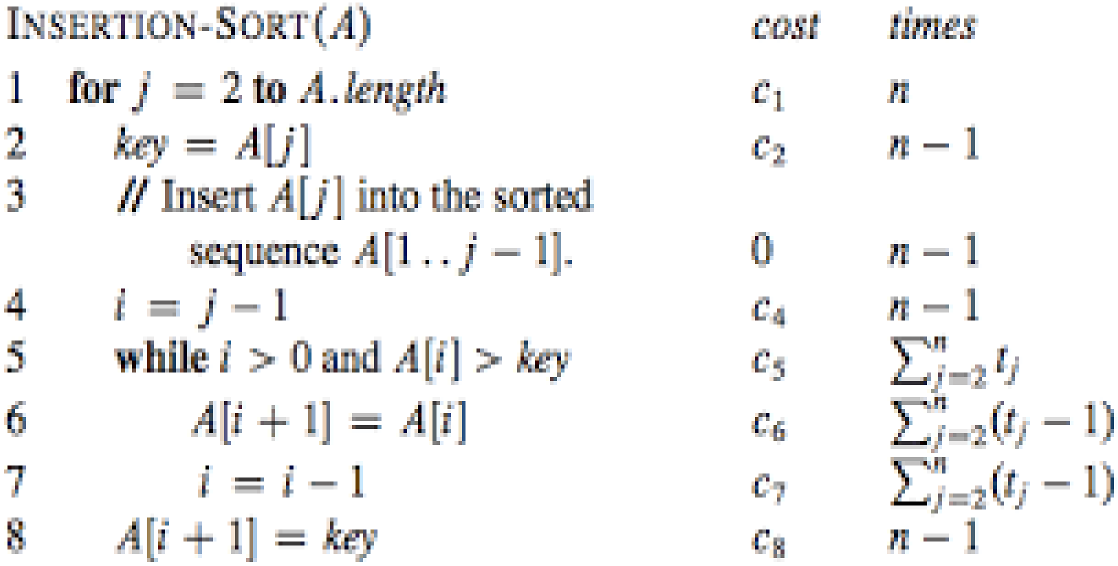

Regarding lab program list - raje
Not Found
Python Program for Insertion Sor
Python | Get a list as input from
Online Python Compiler (Interpre
programiz.com/python-programming/online-compiler/
Paused
Python Online Compiler
SHARE YOUR PODCAST ANYWHERE
SOUNDCLOUD
Get Started
Send Feedback
main.py
Run
Shell
javascript:void(0);
Type here to search
ENG
21:36
01-07-2020

```python
# Online Python compiler (interpreter) to run Python online.
# Write Python 3 code in this online editor and run it.
def insertionSort(arr):
    for i in range(1, len(arr)):
        key = arr[i]
        j=i-1
        while j >= 0 and key < arr[j] :
            arr[j + 1] = arr[j]
            j-=1
        arr[j + 1] = key

arr = []
n=int(input('input size of array'))
for i in range(0, n):
    ele = int(input())
    arr.append(ele) # adding the element
insertionSort(arr)
for i in range(len(arr)):
    print ("% d" % arr[i])
```

```
input size of array5
12
21
32
34
54
 12
 21
 32
 34
 54
>>>
```

Complexity Analysis



```
INSERTION-SORT(A)                                  cost    times
for $j = 2$ to $A.length$                          $c_1$   $n$
    $key = A[j]$                                   $c_2$   $n-1$
    // Insert $A[j]$ into the sorted
           sequence $A[1..j-1]$.                   0       $n-1$
    $i = j-1$                                      $c_4$   $n-1$
    while $i > 0$ and $A[i] > key$                 $c_5$   $\sum_{j=2}^{n} t_j$
        $A[i+1] = A[i]$                            $c_6$   $\sum_{j=2}^{n} (t_j - 1)$
        $i = i-1$                                  $c_7$   $\sum_{j=2}^{n} (t_j - 1)$
    $A[i+1] = key$                                 $c_8$   $n-1$
```

Thank You