NAME: ASIF ERFAN KHAN

ROLL NUMBER: 546

COURSE: MSc CS

SUBJECT: ALGORITHM

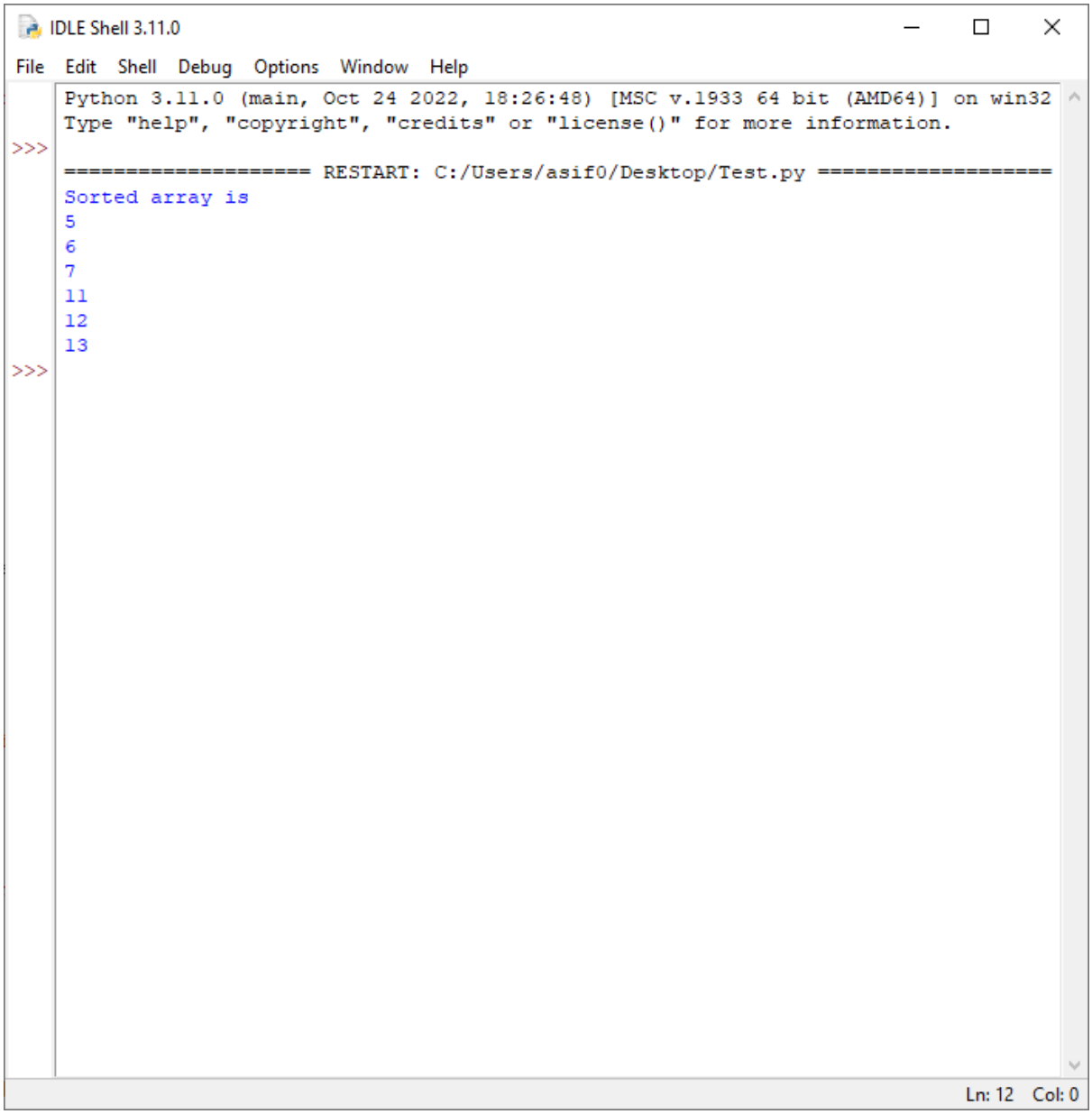
TOPIC: HEAP SORT ALGORITHM

PRACTICAL 2

```
# Python program for implementation of heap Sort


# To heapify subtree rooted at index i.

# n is size of heap



def heapify(arr, n, i):

        largest = i # Initialize largest as root

        l = 2 * i + 1 # left = 2*i + 1

        r = 2 * i + 2 # right = 2*i + 2


# See if left child of root exists and is

# greater than root


        if l < n and arr[i] < arr[l]:

                largest = l


# See if right child of root exists and is

# greater than root


        if r < n and arr[largest] < arr[r]:

                largest = r


# Change root, if needed


        if largest != i:

                (arr[i], arr[largest]) = (arr[largest], arr[i]) # swap


# Heapify the root.
```

```
                heapify(arr, n, largest)


# The main function to sort an array of given size

def heapSort(arr):
    n = len(arr)

# Build a maxheap.
# Since last parent will be at ((n//2)-1) we can start at that location.

    for i in range(n // 2 - 1, -1, -1):
        heapify(arr, n, i)

# One by one extract elements

    for i in range(n - 1, 0, -1):
        (arr[i], arr[0]) = (arr[0], arr[i]) # swap
        heapify(arr, i, 0)



# Driver code to test above

arr = [12, 11, 13, 5, 6, 7, ]
heapSort(arr)
n = len(arr)
print('Sorted array is')
for i in range(n):
    print(arr[i])

# This code is contributed by Mohit Kumra
```

OUTPUT:


IDLE Shell 3.11.0
File Edit Shell Debug Options Window Help
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:/Users/asif0/Desktop/Test.py ==========
Sorted array is
5
6
7
11
12
13
>>>
Ln: 12 Col: 0