Algorithm Practical II

Aim: WAP to demonstrate merge sort Algorithm using divide and conquer technique

Algorithm

1) Start
2) take merge functions with parantes arr, l, m and r
3) initalize variables n1 and n2
4) create arrays L and R
5) Use for loop to copy data in array
6) merge the arrays using while loop
7) using mergesort a function for the sub-array into sort
8) Take the values arr in arr variables to sort
9) print the sorted array
10) Stop

Writeups

→ Merge Sort
→ It is a divide and conquer algorithm based on the idea of breaking down a list into several sub-lists until each sublist consists of a single element and merging those sublists in a manner that results into sort lest

```python
print ("Neeraj Appari S073")
def merge(arr, l, m, r):
    n1 = m - l + 1
    n2 = r- m

    # create temp arrays
    L = [0] * (n1)
    R = [0] * (n2)

    # Copy data to temp arrays L[] and R[]
    for i in range(0 , n1):
        L[i] = arr[l + i]

    for j in range(0 , n2):
        R[j] = arr[m + 1 + j]

    # Merge the temp arrays back into arr[l..r]
    i = 0      # Initial index of first subarray
    j = 0      # Initial index of second subarray
    k = l      # Initial index of merged subarray

    while i < n1 and j < n2 :
        if L[i] <= R[j]:
            arr[k] = L[i]
            i += 1
        else:
            arr[k] = R[j]
            j += 1
        k += 1

    # Copy the remaining elements of L[], if there
    # are any
```

```
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020, 22:20:19) [MS
C v.1925 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more i
nformation.
>>>
===== RESTART: E:/fffiiles/college pracs and projects/Algorit
hm/prac 11.py =====
Neeraj Appari S073
Given array is
12
11
13
2
6
9
18


Sorted array is
2
6
9
11
12
13
18
>>>
```

prac 11.py - E:/fffiiles/college pracs and projects/Algorithm/prac 11.py (3.8.3)

File Edit Format Run Options Window Help

```python
    while j < n2:
        arr[k] = R[j]
        j += 1
        k += 1

# l is for left index and r is right index of the
# sub-array of arr to be sorted
def mergeSort(arr,l,r):

    if l < r:

        # Same as (l+r)//2, but avoids overflow for
        # large l and h
        m = (l+(r-1))//2

        # Sort first and second halves
        mergeSort(arr, l, m)
        mergeSort(arr, m+1, r)
        merge(arr, l, m, r)


# Driver code to test above
arr = [12, 11, 13, 2, 6, 9, 18]
n = len(arr)
print ("Given array is")
for i in range(n):
    print ("%d" %arr[i]),

mergeSort(arr,0,n-1)
print ("\n\nSorted array is")
for i in range(n):
    print ("%d" %arr[i]),
```

Python 3.8.3 Shell

File Edit Shell Debug Options Window Help

```
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020, 22:20:19) [MS
C v.1925 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more i
nformation.
>>>
===== RESTART: E:/fffiiles/college pracs and projects/Algorit
hm/prac 11.py =====
Neeraj Appari S073
Given array is
12
11
13
2
6
9
18


Sorted array is
2
6
9
11
12
13
18
>>>
```

Type here to search