<u>Dashboard</u> / <u>My courses</u> / <u>CS23331-DAA-2023-CSE</u> / <u>Greedy Algorithms</u> / <u>4-G-Array Sum max problem</u>

Started on	Thursday, 29 August 2024, 10:29 AM
State	Finished
Completed on	Thursday, 29 August 2024, 10:41 AM
Time taken	11 mins 55 secs
Marks	1.00/1.00
Cl .	40.00 + - (10.00 (40.00))

```
Question 1
Correct
Mark 1.00 out of 1.00
```

Given an array of N integer, we have to maximize the sum of arr[i] * i, where i is the index of the element (i = 0, 1, 2, ..., N). Write an algorithm based on Greedy technique with a Complexity O(nlogn).

Input Format:

First line specifies the number of elements-n

The next n lines contain the array elements.

Output Format:

Maximum Array Sum to be printed.

Sample Input:

5

25340

Sample output:

40

```
Answer: (penalty regime: 0 %)
      #include <stdio.h>
       void merge(int arr[], int p, int q, int r);
      void mergeSort(int arr[], int 1, int r);
   5 v int main(){
           int n;
scanf("%d",&n);
   6
   7
   8
           int arr[n];
   9,
           for(int i=0;i<n;i++){</pre>
  10
                scanf("%d",&arr[i]);
  11
  12
           mergeSort(arr, 0,n-1);
  13
           int s=0;
  14
           for(int i=0;i<n;i++){</pre>
  15
                  s+=arr[i]*i;
  16
           printf("%d",s);
  17
  18
  19
  20
  21 void merge(int arr[], int p, int q, int r) {
           int n1 = q - p + 1;
int n2 = r - q;
  22
  23
  24
           int L[n1], M[n2];
  25
           for (int i = 0; i < n1; i++)
  26
                L[i] = arr[p + i];
           for (int j = 0; j < n2; j++)
  27
  28
               M[j] = arr[q + 1 + j];
  29
           int i, j, k;
  30
           i = 0;
  31
           j = 0;
  32
           k = p;
  33
           while (i < n1 && j < n2)
  34 ,
  35
                if (L[i] <= M[j])</pre>
  36 ▼
                {
  37
                    arr[k] = L[i];
  38
  39
                }
  40
                else
  41
                    arr[k] = M[j];
  42
  43
                    j++;
  44
  45
                k++;
  46
```

```
47

48 v

49

50

51

52

47

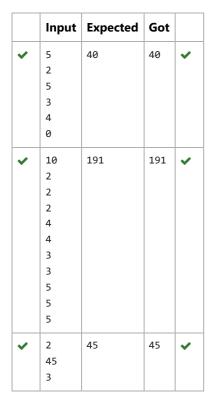
while (i < n1)

{

arr[k] = L[i];

k++;

k++;
```



Passed all tests! 🗸

Correct

Marks for this submission: 1.00/1.00.

◄ 3-G-Burger Problem

Jump to...

5-G-Product of Array elements-Minimum ►