

[Dashboard](#) / [My courses](#) / [CS23331-DAA-2023-CSE](#) / [Greedy Algorithms](#) / [4-G-Array Sum max problem](#)

Started on	Thursday, 22 August 2024, 11:23 AM
State	Finished
Completed on	Thursday, 22 August 2024, 11:43 AM
Time taken	20 mins 2 secs
Marks	1.00/1.00
Grade	10.00 out of 10.00 (100%)

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 ($i = 0, 1, 2, \dots, N$). Write an algorithm based on Greedy technique with a Complexity $O(n \log n)$.

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

2 5 3 4 0

Sample output:

40

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 void mergesort(int arr[], int left, int right, int mid);
3
4 void merge(int arr[], int left, int right){
5     if(left<right){
6         int mid= left + (right-left)/2;
7         merge(arr, left, mid);
8         merge(arr, mid+1, right);
9         mergesort(arr, left, right, mid);
10    }
11 }
12
13 void mergesort(int arr[], int left, int right, int mid){
14     int n1 = mid-left+1;
15     int n2 = right-mid;
16     int a[n1],b[n2];
17
18     for(int i=0 ; i<n1 ; i++){
19         a[i] = arr[left+i];
20     }
21     for(int i=0 ; i<n2 ; i++){
22         b[i] = arr[mid+1+i];
23     }
24
25     int aptr=0, bptr=0, cptr=left;
26     while(aptr<n1 && bptr<n2){
27         if(a[aptr] < b[bptr]){
28             arr[cptr] = a[aptr];
29             aptr++;
30         }
31         else{
32             arr[cptr] = b[bptr];
33             bptr++;
34         }
35         cptr++;
36     }
37     while(aptr < n1){
38         arr[cptr] = a[aptr];
39         cptr++;
40         aptr++;
41     }
42     while(bptr < n2){
43         arr[cptr] = b[bptr];
44         cptr++;
45         bptr++;
46     }
47 }
48
49 int main(){
50     int n;
51     scanf("%d",&n);
52     int arr[n];

```

	Input	Expected	Got	
✓	5 2 5 3 4 0	40	40	✓
✓	10 2 2 2 4 4 3 3 5 5 5	191	191	✓
✓	2 45 3	45	45	✓

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 ▶