



Started on	Sunday, 31 August 2025, 10:14 AM
State	Finished
Completed on	Sunday, 31 August 2025, 10:14 AM
Time taken	13 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 two arrays `array_One[]` and `array_Two[]` of same size `N`. We need to first rearrange the arrays such that the sum of the product of pairs (1 element from each) is minimum. That is $\text{SUM}(A[i] * B[i])$ for all `i` is minimum.

For example:

Input	Result
3	28
1	
2	
3	
4	
5	
6	

Answer: (penalty regime: 0 %)

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  // Ascending sort
5  int compareAsc(const void *a, const void *b) {
6      return (*(int *)a - *(int *)b);
7  }
8
9  // Descending sort
10 int compareDesc(const void *a, const void *b) {
11     return (*(int *)b - *(int *)a);
12 }
13
14 int main() {
15     int n;
16     scanf("%d", &n);
17
18     int array_One[n], array_Two[n];
19
20     // Read array_One
21     for (int i = 0; i < n; i++) {
22         scanf("%d", &array_One[i]);
23     }
24
25     // Read array_Two
26     for (int i = 0; i < n; i++) {
27         scanf("%d", &array_Two[i]);
28     }
29
30     // Sort array_One ascending
31     qsort(array_One, n, sizeof(int), compareAsc);
32
33     // Sort array_Two descending
34     qsort(array_Two, n, sizeof(int), compareDesc);
35
36     // Compute the minimal product sum
37     int result = 0;
38     for (int i = 0; i < n; i++) {
39         result += array_One[i] * array_Two[i];
40     }
41
42     printf("%d\n", result);
43
44     return 0;
45 }
```

	Input	Expected	Got	
✓	3 1 2 3 4 5 6	28	28	✓
✓	4 7 5 1 2 1 3 4 1	22	22	✓
✓	5 20 10 30 10 40 8 9 4 3 10	590	590	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)