

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 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 void bubbleSort(int arr[], int n) {
3     for (int i = 0; i < n - 1; i++) {
4         for (int j = 0; j < n - i - 1; j++) {
5             if (arr[j] > arr[j + 1]) {
6                 int temp = arr[j];
7                 arr[j] = arr[j + 1];
8                 arr[j + 1] = temp;
9             }
10        }
11    }
12 }
13 void bubbleSort2(int arr[], int n) {
14     for (int i = 0; i < n - 1; i++) {
15         for (int j = 0; j < n - i - 1; j++) {
16             if (arr[j] < arr[j + 1]) {
17                 int temp = arr[j];
18                 arr[j] = arr[j + 1];
19                 arr[j + 1] = temp;
20             }
21        }
22    }
23 }
24 int main(){
25     int n,sum=0;
26     scanf("%d",&n);
27     int arr1[n],arr2[n];
28     for(int i=0;i<n;i++)scanf("%d",&arr1[i]);
29     for(int i=0;i<n;i++)scanf("%d",&arr2[i]);
30     bubbleSort(arr1,n);
31     bubbleSort2(arr2,n);
32     for(int j=0;j<n;j++){
33         sum+=(arr1[j]*arr2[j]);
34     }
35     printf("%d",sum);
36 }
```

	Input	Expected	Got	
✓	3	28	28	✓
	1			
	2			
	3			
	4			
	5			
	6			
✓	4	22	22	✓
	7			
	8			