Flowers

You and your $K-1$ friends want to buy N flowers. Flower number i has cost c_i. Unfortunately the seller does not want just one customer to buy a lot of flowers, so he tries to change the price of flowers for customers who have already bought some flowers. More precisely, if a customer has already bought x flowers, he should pay $(x+1) \times c_i$ dollars to buy flower number i.
You and your $K-1$ friends want to buy all N flowers in such a way that you spend the least amount of money. You can buy the flowers in any order.

Input:

The first line of input contains two integers N and $K(K <= N)$. The next line contains N space separated positive integers $c_1, c_2, \ldots, c_N$.

Output:

Print the minimum amount of money you (and your friends) have to pay in order to buy all N flowers.

Constraints

$1 \leq N, K \leq 100$
Any c_i is not more than 10^6
Result is guaranteed to be less than 2^{31}

Sample input #00

```
3 3
2 5 6
```

Sample output #00

```
13
```

Sample input #01

```
3 2
2 5 6
```

Sample output #01

```
15
```

Explanation :
Sample Case #00: In this example, all of you should buy one flower each. Hence, you'll have to pay 13 dollars.

Sample Case #01: Here one of the friend buys first two flowers in decreasing order of their price. So he will pay `(0+1)*5 + (1+1)*2 = 9`. And other friend will buy the costliest flower of cost `6`. So total money need is `9+6=15`.

Related Topics

Greedy Technique

Sorting

Submissions: 13036

Max Score: 35

Difficulty: Moderate

More

Current Buffer (saved locally, editable) Java 8

```
1  import java.io.*;
2  import java.util.*;
3
4  public class Solution {
5
6      public static void main(String[] args) {
7          /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named
   Solution. */
8      }
9  }
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code