



Greedy Florist ★

[Problem](#)[Submissions](#)[Leaderboard](#)[Editorial](#)[Topics](#)

A group of friends want to buy a bouquet of flowers. The florist wants to maximize his number of new customers and the money he makes. To do this, he decides he'll multiply the price of each flower by the number of that customer's previously purchased flowers plus **1**. The first flower will be original price, $(0 + 1) \times \text{original price}$, the next will be $(1 + 1) \times \text{original price}$ and so on.

Given the size of the group of friends, the number of flowers they want to purchase and the original prices of the flowers, determine the minimum cost to purchase all of the flowers. The number of flowers they want equals the length of the **c** array.

Example

c = [1, 2, 3, 4]

k = 3

The length of **c** = 4, so they want to buy 4 flowers total. Each will buy one of the flowers priced [2, 3, 4] at the original price. Having each purchased **x** = 1 flower, the first flower in the list, **c**[0], will now cost $(\text{current purchase} + \text{previous purchases}) \times c[0] = (1 + 1) \times 1 = 2$. The total cost is $2 + 3 + 4 + 2 = 11$.

Function Description

Complete the getMinimumCost function in the editor below.

getMinimumCost has the following parameter(s):

- int c[n]: the original price of each flower
- int k: the number of friends

Returns

- int: the minimum cost to purchase all flowers

Input Format

The first line contains two space-separated integers **n** and **k**, the number of flowers and the number of friends.

The second line contains **n** space-separated positive integers **c**[*i*], the original price of each flower.

Constraints

- $1 \leq n, k \leq 100$
- $1 \leq c[i] \leq 10^6$
- $\text{answer} < 2^{31}$
- $0 \leq i < n$

Sample Input 0

```
3 3
2 5 6
```

Sample Output 0

```
13
```

Explanation 0

There are **n** = 3 flowers with costs **c** = [2, 5, 6] and **k** = 3 people in the group. If each person buys one flower, the total cost of prices paid is $2 + 5 + 6 = 13$ dollars. Thus, we print **13** as our answer.

Sample Input 1

```
3 2
2 5 6
```

Sample Output 1

```
15
```

Explanation 1

There are $n = 3$ flowers with costs $c = [2, 5, 6]$ and $k = 2$ people in the group. We can minimize the total purchase cost like so:

1. The first person purchases **2** flowers in order of decreasing price; this means they buy the more expensive flower ($c_1 = 5$) first at price $p_1 = (0 + 1) \times 5 = 5$ dollars and the less expensive flower ($c_0 = 2$) second at price $p_0 = (1 + 1) \times 2 = 4$ dollars.
2. The second person buys the most expensive flower at price $p_2 = (0 + 1) \times 6 = 6$ dollars.

We then print the sum of these purchases, which is $5 + 4 + 6 = 15$, as our answer.

Sample Input 2

```
5 3
1 3 5 7 9
```

Sample Output 2

```
29
```

Explanation 2

The friends buy flowers for **9, 7** and **3, 5** and **1** for a cost of $9 + 7 + 3 * (1 + 1) + 5 + 1 * (1 + 1) = 29$.

[Change Theme](#)

JavaScript (Node.js)



```

25 }
26
27 // Complete the getMinimumCost function below.
28 function getMinimumCost(k, c) {
29     let price = 0;
30     let prevPurchases = 1;
31     let friends = k;
32
33     c.sort((a, b) => a - b);
34
35     for (let i = c.length - 1; i >= 0; i--) {
36         price += prevPurchases * c[i];
37         friends--;
38         if (friends === 0) {
39             prevPurchases++;
40             friends = k;
41         }
42     }
43     return price;
44 }
45
46 function main() {
47     const ws = fs.createWriteStream(process.env.OUTPUT_PATH);
48

```

Line: 43 Col: 18

☒ Upload Code as File
 ☐ Test against custom input

Run Code

Submit Code

Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

✔ Sample Test case 0

✔ Sample Test case 1

✔ Sample Test case 2

Input (stdin)

Download

1	3 3
2	2 5 6

Your Output (stdout)

1	13
---	----

Expected Output

Download

1	13
---	----