Problem A. Shopping Fever

Purchasing two items together is never needed: you can instead purchase each of them separately and get the same discount.

Purchasing more than three items together is also never needed: you can instead purchase just three of them together and the rest separately and you will pay *at most* the same as for the original purchase.

Thus, there is always an optimal solution in which we purchase some items in groups of three and the rest individually.

In Subtask 1 it is therefore sufficient to check only two cases: either we purchase all three items together, or we purchase each item separately.

In Subtask 2 we have $q = 0$ and thus the only way to get a discount is to get some items for free. In order to get an item for free, we need to buy it together with (at least) two items that are at least equally expensive. From this observation it can easily be shown that the optimal solution in this case is to sort the item costs in descending order and purchase the items in that order, in groups of three.

In Subtask 3 we have $q = 40\%$. Note that with promotion #1 we can save at most one third of the total price of the purchased items (which is about 33.3%). Hence, whenever q is at least 34, the optimal solution is to purchase all items one at a time, always getting the q percent discount.

We are left with cases where q is between 1 and 33, inclusive. In these cases we might be tempted to solve the task greedily, too, but simple greedy strategies aren't always optimal in these cases.

For example, suppose the discount is $q = 10$ percent and the item costs are $1000, 900, 100, 100, 100$. If you purchase $1000 + 900 + 100$ together as in the solution to Subtask 2, you will get the item worth 100 for free, but for these three items you only saved five percent of their total price. This is because the third item was much cheaper than the other two. It would have been better to buy each of them separately. But that's still not optimal. For this test case, the optimal solution is to buy the items worth 1000 and 900 separately, and to purchase the other three items together to save 100 on them.

Let's think some more about the structure of optimal solutions. In Subtask 2 it was optimal to always buy three consecutive items together. Can we generalize this somehow?

Sure we can. Suppose the item costs are $p_1 \geq p_2 \geq \dots p_{n-1} \geq p_n$. We now claim that we can always construct an optimal solution such that whenever we buy three items together, they are at consecutive positions in this sorted sequence.

First, suppose we already decided which subset of items will be purchased in groups of three and which ones will be purchased separately. For those purchased in groups of three we have the same situation as in Subtask 2: we know that the optimal solution is to form the groups of three in the order of costs. Thus, it is enough to look for a solution in which the groups of three don't overlap in the sorted order of costs.

Second, suppose we are purchasing items i, j, k together and $k \neq i+2$. Instead of doing so, we can purchase $k-2, k-1, k$ together and each element in the range $[i, k-3]$ separately. We will still get element k for free, and in the new solution the individual items that get the discount are at least as expensive as in the old solution.

At this moment we therefore know that there is always an optimal solution in which each purchase of three items is a purchase of three consecutive items in the sorted order. In other words, in this optimal solution we take the sorted sequence of items and divide it into blocks of sizes 1 and 3.

The observation we just made allows us to find the best solution of this form using *dynamic programming*. We will process the prices in decreasing order. In each step i we will compute the minimum cost C_i we have to pay in order to buy the items with prices $p_1, \dots p_i$. We start with $C_0 = 0$. In step i we consider the problem of purchasing the first i items optimally. There are two options: either item i is purchased alone, or (for $i \geq 3$) the triple $i-2, i-1, i$ is purchased together.

The cheapest solution of the first type has cost $C_{i-1} + (100 - q) * p_i/100$ because we get item i at a discount and we need to purchase the remaining $i-1$ items as cheaply as possible. The cheapest solution of the second type has cost $C_{i-3} + p_{i-2} + p_{i-1}$. The optimal solution is obtained by picking the cheaper option, so C_i equals the minimum of those two costs.

The final answer is the value C_n. The total time complexity of this solution is $O(n \log n)$ due to sorting, the memory complexity is $O(n)$.