ECOO '13 R1 P4 - Coupon Day

Time limit: 13.0s **Memory limit:** 256M

It's coupon day at Panther Redirect. Customers have been collecting coupons all year, and today is the day they get to use them. During this special sale day, there's a limit of 10 purchases per customer and they are allowed to bring up to 10 coupons with them to the counter. Each coupon can be applied to a maximum of 1 item, and each item can have a maximum of 1 coupon applied to it. The cashier will scan the price codes and the coupons and will help the customers decide how to use their coupons to maximize their savings.

There are 7 coupon types available. The $\$5$, $\$10$, and $\$50$ coupons entitle the customers to a flat discount before tax is applied (if the item is worth less than the coupon, they get it for free). The 10% and 20% coupons entitle the customer to a percentage discount before tax. The `TAX` coupon entitles the customer to have the item without paying any HST. Finally, the `BOGO` coupon (maximum of 1 per customer) allows the user to buy one item at full price and get a second item of equal or lesser price for free. Note that neither of the items involved in the `BOGO` can have other coupons applied to them. The 13% HST is calculated separately on the unrounded price of each item after the coupon is applied. The after tax price for each item is rounded to the nearest cent after tax has been applied. These final prices are added together to get the total purchase price.

The input will contain 5 test cases. The first line of each test case contains an integer N indicating the number of purchase items $(1 \leq N \leq 10)$. This is followed by the N prices P_i in dollars and cents, each on a separate line $(0.0 < P_i \leq 100.0, 1 \leq i \leq N)$. The next line contains an integer M, indicating the number of coupons $(1 \leq M \leq 10)$. This is followed by the M coupon names, each on a separate line.

Write a program that finds the best way to apply the coupons for each customer (the best way being the way that yields the lowest total price according to the rules and restrictions applied above) and then states the final price exactly as shown in the sample output below, always showing two decimal places. The program must terminate within the time limit set out in the general contest rules.

Sample Input

```
3
74.54
19.8
69.99
10
BOGO
20%
$50
BOGO
20%
TAX
20%
$5
$5
10%
9
93.43
13.69
17.02
1.94
6.52
65.55
8.36
83.2
0.11
10
$5
$10
$10
TAX
$5
20%
BOGO
BOGO
TAX
BOGO
4
88.17
43.18
67.14
2.51
5
20%
20%
$50
```

```
TAX
$50
```

Sample Output

```
The best price is $84.23
The best price is $184.51
The best price is $101.35
```