Minimizing Cost 1

Problem Submissions Leaderboard

You are given a keypad with ten buttons representing digits from **[0,9]**, when pressed, the corresponding digit appears on the screen. There is an additional button called **Addition Modulo 10** which can be used to replace the last two digits **a**, **b** with **x** according to the following equation.

$x=(a+b) mod 10$

If there are less than two digits on the screen, **Addition Modulo 10** does not work. The cost of pressing any digit is **Ci** for $i = 0, 1, ..., 9$ and the cost of pressing **Addition Modulo 10** button is always **0**.

If you are given the cost of pressing each button, find the minimum cost of feeding a given target number **S** into the screen using a sequence of button presses.

Input Format

- The first line of the input contains the number of test cases **T**.
- The first line of each test case contains 10 space-separated integers **Ci**, the cost of pressing buttons from 0 to 9. The second line contains the length of the target number **S** and the third line contains the target number itself.

Constraints

- $1 \leq T \leq 1000$
- $0 \leq Ci \leq 1000$
- $1 \leq |S| \leq 1000$

Output Format

For each test case, print the minimum cost of feeding the target number into screen.

Sample Input 0

```
3
3 2 2 3 2 1 1 2 3 3
3
171
3 2 3 1 1 1 1 3 1 2
2
16
3 3 3 1 3 1 1 2 3 2
2
43
```

Sample Output 0

```
6
3
4
```

Contest ends in 2 hours

Submissions: 155
Max Score: 100
Difficulty: Medium

Rate This Challenge:

More

Current Buffer (saved locally, editable)

C++14

```
#include <map>
#include <set>
#include <list>
#include <cmath>
#include <ctime>
#include <deque>
#include <queue>
#include <stack>
#include <string>
#include <bitset>
#include <cstdio>
#include <limits>
#include <vector>
#include <climits>
#include <cstring>
#include <cstdlib>
#include <fstream>
#include <numeric>
#include <sstream>
#include <iostream>
#include <algorithm>
#include <unordered_map>

using namespace std;
int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}
```

Line: 1 Col: 1

Upload Code as File

Test against custom input

Run Code

Submit Code