ETH Zurich
Institut für Theoretische Informatik
Prof. Dr. Angelika Steger

HS 2015
Exercise Sheet 1

ACM Programming Challenges Lab

Exercise 5 – *Checking Change*

Description You are working for company producing vending machines and you are investigating some problem reports from your customers. In certain countries the machines seem to give change using too many coins instead of the minimum possible number. In other countries on the other hand it seems that the machine is unable to hand out the correct change in some cases, and always gives back slightly more than what is necessary. This seems strange to you, so you try to find out for any given set of coins that your machine accepts and some common return values what the optimal number of coins should be.

Input The first line of the input contains n, the number of currencies you wish to test. Each test case starts with one line containing two numbers c_i and m_i, $1 \leq i \leq n$. The integer c_i, $1 \leq c_i \leq 100$ is the amount of different coin-values that the machine accepts in that currency, and m_i is the number of sample return values you would like to test. The next line contains c_i different integers, sorted in ascending order, which represent the coin values for the currency. Then m_i lines follow, each containing an integer t in the range $1 \leq t < 10000$ representing the change your machine needs to hand out. These numbers are in the same unit as the given coin values.

Output For each test case i you should output m_i lines, each with the minimum number of coins necessary to hand out the given change. If there is no possible way of constructing the change value you should output "not possible".

Sample input

```
3
3 1
1 2 5
17
1 1
3
2
2 3
2 3
3
5
1
```

Sample output

```
4
not possible
1
2
not possible
```