Expression Puzzle Problem

Introduction

Implement a dynamic programming algorithm that solves optimally the expression puzzle problem. Given a set of digits $S = \{S_i\}_{i=1..k}$ and an integer N, the expression puzzle problem is finding a string consisting of characters from S (you can repeat a character as many times as you need) and the special symbols "+" and "*" (also you can repeat them as many times as you need) such that an arithmetic evaluation of the resulting expression yields the number N. For example, if $S = \{2, 9\}$, N=229 can be obtained by creating a string by concatenating the digits 2, 2 again followed by 9: "229". N= 11 can be obtained by the string "2+9", N =49 can be obtain using the string "9+2*9+22", etc. An optimal solution is a solution that has minimal character length (i.e. any other solution string has more or equal number of characters). For example, for N=22, both "2*9+2+2" and "22" are valid puzzle solution, but only the latter is optimal. Note that for some puzzles many optimal solutions may exist, and some other puzzles may have no valid solution.

Specifications

The input is specified in a file whose name is the first argument of the program. The first line contains an integer M specifying how many datasets are in the file. The reminder of the file encodes the datasets. Each dataset is encoded in one line. It starts with an integer K that indicates how many elements are in the set S, followed by the actual digits in S (you can assume that the digits do not repeat). The last number in the line is the integer number N. Note that K, $\{S_i\}_{i=1..k}$ and N are separated by spaces. N<20000.

Here is an example:

6
2 2 9 229
2 2 9 11
2 2 9 729
2 2 9 49
3 1 4 7 21
2 4 7 6

The output is a file called whose name is the second argument of the program. Each line encodes the results of each test case. If a solution exists, the algorithm should output the length of the optimal sequence of the puzzle, followed by the string of the expr4ssion itself. If none solution exists, the program should output the character "N".

For example, one output corresponding to the input above is as follows (multiple correct outputs are possible):

3 229
3 2+9
5 9*9*9
8 2+9+9+29

4 4+17
N

Submission

You have to implement your program in plain C++ 11 in a file called *main.cpp* that has no dependency other than the standard C/C++ libraries available in a standard Linux system. The code should compile using the command *g++ -std=c++11 main.cpp*.