Sereja and Functions (Challenge)

Problem Code: **SEAFUNC**

Submit (/SEPT17/submit/SEAFUNC)

My Submissions (/SEPT17/status/SEAFUNC,jacobian8162) All Submissions (/SEPT17/status/SEAFU

Successful Submissions +

Sereja is stuck with a problem and asks for your help. You readily agree to help Sereja and he poses the following problem to you.

Sereja has a matrix **A** of size **N x N**. Each cell in row **i** and column **j** contains a number **A[i][j]**. Every element of the matrix can either be **0** or **1**. Initially, all the elements are zero.

Sereja informs you that he only likes the functions of the form

$$f(x) = \frac{x^3 a_1}{a_2} + \frac{x^2 b_1}{b_2} + \frac{x\, c_1}{c_2} + d$$

Here the divison is integer divison, i.e. x/y means floor(x/y).

Sereja can choose a set of integers a_1, a_2, b_1, b_2, c_1, c_2, **d** and two other integers **l, r** and **applies** the function **f** to the matrix as follows. He will substitute 1 in the every cell **(i, j)** where **l ≤ i ≤ r** and **j = f(i)**.

Sereja gives you a matrix **A** after applying a certain number of functions to it. He asks you to find the minimal amount of functions such that after applying them Sereja will get the matrix A. You need not create the matrix **A** exactly, you are allowed to have at most **100 distinct cells** in the matrix obtained after your operations and the matrix **A**. Your task is to minimize the number of such functions applied.

Input

The first line of input contains an integer **T** denoting the number of test cases.

First line of each test case contains an integer **N**. Each of the next **N** lines contains **N** digits (zero or one) without spaces denoting the matrix **A**.

Output

For each test case in first line output an integer corresponding to the number of functions **Q**.

Next **Q** lines should contain information about functions. Each lines should contain set of integers a_1, a_2, b_1, b_2, c_1, c_2, **d**, l, r.

The following constraints must be satisfied for the output.

- **0 ≤ Q ≤ N*N**
- **-N ≤ a_1, b_1, c_1, d ≤ N**
- **1 ≤ a_2, b_2, c_2 ≤ N**
- **1 ≤ l ≤ r ≤ N**

Constraints

- **1 ≤ T ≤ 20**
- **1 ≤ N ≤ 100**

Tests generation

There will be **10** official tests. During contest you will be able to get your score on first **2** tests. After contest there will be rejudge on full tests set.

In every test, **T** is equal to **20** and **N** is chosen randomly in the range **[95, 100]**. Further for each test case integer **K** is chosen:

- For first **5** tests, **K** is chosen randomly in range [1, 5]
- For next **5** tests, **K** is chosen randomly in range [50, 100]
- For next **5** tests, **K** is chosen randomly in range [500, 1000]
- For next **5** tests, **K** is chosen randomly in range [1000, 3000]

After **K** is chosen we generate **K** functions in next way:

- With probability **1/3**, $a_1 = b_1 = 0$
- With probability **2/3**, $a_1 = 0$
- For all other numbers, the way of generation is hidden

Scoring

You will receive a WA if the operation doesn't satisfy the output constraints specified in the statement or the number of distinct cells in the final matrix after applying all the operations exceeds 100.

Lets **S** denote the sum of **Q / (N * N + 1 - ONES)** (where **ONES** is number of digit **1** in given matrix) for all the test cases. Your score will be to **S** which you should try to minimize.

Example

```
Input:
1
5
00010
00100
01000
10000
11111

Output:
2
0 1 0 1 -1 1 5 1 4
0 1 0 1 0 1 5 1 5
```

Author:	6★ sereja (/users/sereja)
Date Added:	21-11-2015
Time Limit:	1 secs
Source Limit:	50000 Bytes
Languages:	ADA, ASM, BASH, BF, C, C99 strict, CAML, CLOJ, CLPS, CPP 4.3.2, CPP 6.3, CPP14, CS2, D, ERL, FORT, FS, GO, HASK, ICK, ICON, JAVA, JS, LISP clisp, LISP sbcl, LUA, NEM, NICE, NODEJS, PAS fpc, PAS gpc, PERL, PERL6, PHP, PIKE, PRLG, PYPY, PYTH, PYTH 3.5, RUBY, SCALA, SCM chicken, SCM guile, SCM qobi, ST, TCL, TEXT, WSPC

Comments

The time now is: 01:49:19 PM
Your IP: 106.79.135.29

CodeChef (http://www.codechef.com) - A Platform for Aspiring Programmers

CodeChef was created as a platform to help programmers make it big in the world of algorithms, **computer programming** and **programming contests**. At CodeChef we work hard to revive the geek in you by hosting a **programming contest** at the start of the month and another smaller programming challenge in the middle of the month. We also aim to have training sessions and discussions related to **algorithms**, **binary search**, technicalities like **array size** and the likes. Apart from providing a platform for **programming competitions**, CodeChef also has various algorithm tutorials and forum discussions to help those who are new to the world of **computer programming**.

Practice Section (https://www.codechef.com/problems/easy) - A Place to hone your 'Computer Programming Skills'

Try your hand at one of our many practice problems and submit your solution in a language of your choice. Our **programming contest** judge accepts solutions in over 35+ programming languages. Preparing for coding contests were never this much fun! Receive points, and move up through the CodeChef ranks. Use our practice section to better prepare yourself for the multiple **programming challenges** that take place through-out the month on CodeChef.

Compete (https://www.codechef.com/problems/easy) - Monthly Programming Contests and Cook-offs

Here is where you can show off your **computer programming skills**. Take part in our 10 day long monthly coding contest and the shorter format Cook-off **coding contest**. Put yourself up for recognition and win great prizes. Our **programming contests** have prizes worth up to INR 20,000 (for Indian Community), $700 (for Global Community) and lots more CodeChef goodies up for grabs.

Programming Tools

Online IDE (https://www.codechef.com/ide)
Upcoming Coding Contests (http://www.codechef.com/contests#FurtureContests)
Contest Hosting (http://www.codechef.com/hostyourcontest)
Problem Setting (http://www.codechef.com/problemsetting)
CodeChef Tutorials (http://www.codechef.com/wiki/tutorials)
CodeChef Wiki (https://www.codechef.com/wiki)

Practice Problems

Easy (https://www.codechef.com/problems/easy)
Medium (https://www.codechef.com/problems/medium)
Hard (https://www.codechef.com/problems/Hard)
Challenge (https://www.codechef.com/problems/challenge)
Peer (https://www.codechef.com/problems/extcontest)
School (https://www.codechef.com/problems/school)
FAQ's (https://www.codechef.com/wiki/faq)

Initiatives

Go for Gold (http://www.codechef.com/goforgold)
CodeChef for Schools (http://www.codechef.com/school)
Campus Chapters (http://www.codechef.com/campus_chapter/about)