ADVANCED CATEGORY

UT Dallas Fall High School Programming Contest
November 9th 2013

- Time Allowed: three hours.
- Each team must use only one computer - one of UTD's
- Answer the questions in any order.
- Use Java 1.7, minGW g++, Microsoft Visual Studio 10 C/C++/C#
- Your program source code must be contained in one source file.
- Do not use the "package" construct in your Java code.
- Your programs must read from the named file specified in the problem header, and must output to System.out.
- Do not access the web except to access Java documentation.
- Do not use any recording device containing code, other than UTD's computer.
- Your solutions must be entirely yours, typed by you during the time allowed for the contest.
- As soon as you have solved a problem, submit ONLY the source file via your `PC^2` client.

Scoring

Equal points will be awarded to each question, even though they may not be equally difficult.

In order to break ties, we will use penalty points. The penalty points for a question will be zero if the question is not answered correctly. If a correct submission occurs for a question at time T minutes, then T penalty points will be added, plus 20 penalty points for each incorrect submission for that question.

A Ants

Input Data File: A.txt
Runtime Limit: 3 secs

The ants of a particular colony are in search of food. Unfortunately hidden dangers are all around the colony which makes foraging difficult. There are traps, obstacles, and predators lurking about. Fortunately, the colony has the perfect ant for the job. Max is neither a smart ant nor an efficient ant but he has got blind luck on his side. In all of his wanderings, he has always managed to stay on safe ground and he (eventually) always finds a source of food to report back to the colony.

The problem is that Max rarely takes anything resembling an optimal (shortest) route to find a food source. However, Max can reliably bring back the exact details of the (often winding and convoluted) path that he took to get to the food source. Your job is to help the colony by finding the optimal route located within Max's convoluted directions to allow the colony to forage more efficiently.

Input:
The first integer in the input, n ($1 \leq n \leq 100$), denotes the number of paths to food that Max has reported back. This is followed by a blank line and then descriptions of each of the n paths. Each path description begins with an integer, s ($0 \leq s \leq 60$), which denotes the number of steps taken in the path. The next s lines contain directional steps expressed as upper-case characters N, E, S, and W corresponding to steps taken in the directions north, east, south, and west respectively. Each step moves Max one unit of distance. Max's paths always start at the colony and end at a food source. Between each pair of path descriptions is a blank line.

When searching for an optimal path, the only directional steps that may be taken are ones that have previously been taken by Max, or the same steps in reverse.

Output:
For each given path, give the number of steps found to be in an optimal (shortest) path.

Sample Input	Output for Sample Input
3 8 S E E E N W S S 4 S E N W 3 S E N	4 0 3

B Four Fours

Input Data File: B.txt
Runtime Limit: 3 secs

Write a program which, given an integer n as input, will produce a mathematical expression whose solution is n. The solution is restricted to using exactly four 4s and exactly three of the binary operations selected from the set $\{+, -, *, /\}$. The number 4 is the ONLY number you can use. You are not allowed to concatenate threes to generate other numbers, such as 44 or 444.

For example given $n = 16$, a solution is `4 + 4 + 4 + 4 = 16`. Given $n = 1$, a solution is `4 / 4 * 4 / 4 = 1`. Division is considered truncating integer division, so that $1/4$ is 0 (instead of 0.25). Assume the usual precedence of operations so that $4+4*4 = 20$, not 32. Not all integer inputs have solutions using three 4s with the aforementioned restrictions (consider n = 11).

Input:
Input begins with an integer m ($1 \leq m \leq 100$), indicating the number of test cases that follow. Each of the next m lines contain exactly one integer value for n in the range $-1000 \leq n \leq 1000$.

Output:
For each test case print one line of output containing either an equation using three 3s to reach the target number or the phrase "no solution". Print the equation following the format of the sample output; use spaces to separate the numbers and symbols printed in the equation but not in the case of a unary minus to the right of the equals sign. For example, output "`4 - 4 * 4 = -12`".

If there is more than one equation which evaluates to the target integer, apply the following rules.

(a) Give the thee operator symbols the following integer values:

+	1
−	2
*	3
/	4

(b) concatenate the integer values of three operator symbols in each result,

(c) choose the result that has the lowest operator value.

For example,
"`4 + 4 / 4 / 4`" has operator value 144,
"`4 - 4 / 4 / 4`" has operator value 244, and
"`4 / 4 / 4 + 4`" has operator value 441.
Of these three results for $n = 4$, choose the first.

Sample Input	Output for Sample Input
6	`4 - 4 - 4 / 4 = -1`
-1	`4 + 4 + 4 / 4 = 9`
9	`4 + 4 - 4 - 4  = 0`
0	`4 + 4 - 4 / 4  = 7`
7	no solution
11	`4 + 4 + 4 * 4 = 24`
24	

C Take Two

Input Data File: C.txt
Runtime Limit: 3 secs

Take Two is a game for one player. It is played with a stack of cards each having a numerical value from 1 to 100. The cards are laid out from left to right in a row. At every step, the player is allowed to remove two adjacent cards if the sum of their values is even. The gap is then closed by left-shifting all the cards that are to the right of the gap. The order of the remaining cards is not changed. The game stops when all cards have been removed or when no more cards can be removed. The player wins when all cards are removed. If this is not possible, the player should try to minimize the number of cards remaining.

You are given the initial list of cards, in left-to-right order. Determine the minimum number of cards that remain if the player moves optimally.

Input:
The input consists of one case. The first line contains an integer n ($1 \leq n \leq 1000$) giving the number of cards to follow. The second line contains n integers indicating the card values from left to right. Each card value is in the range 1 to 100.

Output:
Print the minimum number of cards that remain if the player moves optimally.

Sample Input 1	Sample Output 1
10 1 2 3 4 5 6 7 8 9 10	10

Sample Input 2	Sample Output 2
10 1 3 3 4 2 4 1 3 7 1	2

D Pavers

Input Data File: D.txt
Runtime Limit: 3 secs

A path of length n is to be paved with paving stones of lengths $\{1, 2, 3, 4\}$. There is an inexhaustible supply of each size of stones. The customer would like to optimize the aesthetics of her property and asks to see all the possible arrangements of paving stones. The builder is exasperated and wants you to write a program to tell him, for a given length of path, n, how many different arrangements of the paving stones are possible.

Input:
The input file contains multiple test cases. Each test case consists of a single line containing a single integer, n , $1 \leq n \leq 60$. The input is terminated by End of File.

Output:
For each input case, print the case number (beginning with 1) and the number of different arrangements possible with the paving stones.

Sample Input	**Output for Sample Input**
1	Case 1: 1
2	Case 2: 2
4	Case 3: 8
10	Case 4: 401

E Perfect Pitch

Input Data File: E.txt
Runtime Limit: 3 seconds

In modern western music there are 12 notes in an octave. A *semitone* is a raising or lowering of the frequency of a note by the factor $\sqrt[12]{2}$. The notes in the major key of C are named C,D,E,F,G,A,B,C, where the second C has twice the frequency of the first. These 8 notes are separated by *musical intervals* of `tone, tone, semitone, tone, tone, tone, semitone`. Another way of expressing the 12 notes in the key of C Major is: C, C♯, D, D♯, E, F, F♯, G, G♯, A, A♯, B, C (of course, C♯ = $D\flat$).

On the piano the lowest note is known as A0 and the highest, C8. *Middle C* is C4. The frequency of A4 is 440 Hz.

What is the frequency of C♯5?

Given the name of a note and its octave number, compute its frequency.

Input:
The input will comprise a series of lines of up to 50 lines of text, each one containing a note description in the form `D 3`, `C sharp 4`, or `E flat 3`, where the notes are in the key of C major and the octave number is an integer in the range [0,8]. The input will be terminated by the End of File.

Output:
For each note given in the input, output a single line of text giving the frequency of the note, accurate to two decimal places.

Sample Input	Sample Output
`D 3`	`146.83`
`C sharp 4`	`277.18`
`D 4`	`293.66`
`E 5`	`659.26`
`D flat 6`	`1108.73`

F It's Modern Art Man!

Input Data File: F.txt
Run Time Limit 3 sec.

Jaques (his real name is Jack, but he sells more paintings when he pretends to be French) likes to paint geometric shapes. He is in his Circle and Triangle phase. He wants to paint a circle with the largest possible equilateral triangle inside it, then the largest possible circle inside the triangle, and so on, ending up with a tiny dot that represents Nirvana, or Brownwood Texas, or some other place that is likely to sell well in the upscale Dallas art galleries. By rotating the triangles successively by a random angle, Jaques hopes to portray the difficulties of the journey through life, until the awakening occurs.

Jaques hates math, but knows that he must accurately depict the shapes or some smart student from Texas will point out his errors and ruin the spiritual impact of his masterpiece, which he will call, "Journey to the Center of Euclid." Please help him by writing a program to calculate the diameter on the N^{th} circle, the first being the outer circle.

Input:

The first line of the input will contain a single integer, p, $(1 \leq p \leq 20)$, giving the number of lines of input to follow. Then p lines follow, each one containing two integers, S and N, $(1 \leq S,\ N \leq 100)$, where S is the diameter the outer circle and N is the index number of the circle that Jaques must paint next.

Output: For each line of input, output the diameter of the N^{th} circle accurate to 3 decimal places. Print all three places, even if they are all zeros.

Sample Input	Output for Sample Input
2 100 1 100 10	100.000 0.195

G One squared?

Input Data File: G.txt
Run Time Limit 3 sec.

In a square array whose elements are either zero or one, what's the largest square region in which all the elements are equal to one?

Input:
There will be several datasets in the input. The first line of the data contains a single integer P, $(1 \leq P \leq 100)$ giving the number of datasets. P datasets follow.

Each dataset begins with one integer, L $(1 \leq L \leq 40)$ on a single line giving the width and height of the input rectangle Then L lines follow, each containing L characters from the set {'0','1'}.

Output:
For each dataset print a single line of the form, "The largest square contains X ones" where X is the number of ones within the square,
OR,
in the case that the largest square contains exactly one 1, "The largest square contains 1 one".

Sample Input	Output for Sample Input
3 3 011 111 010 4 0111 1111 1111 0111 2 01 10	The largest square contains 4 ones The largest square contains 9 ones The largest square contains 1 one