

CALGARY COLLEGIATE PROGRAMMING CONTEST

2007



University of Calgary

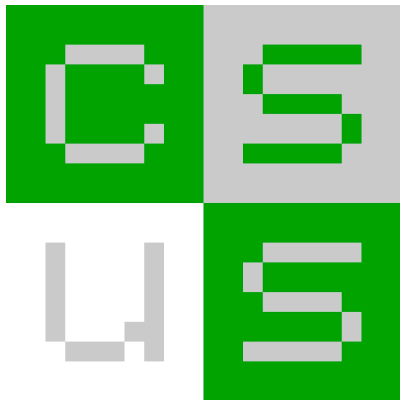
September 17, 2007



EVENT SPONSOR
Online Business Systems

CONTEST ORGANIZER
Sean McIntyre

TECHNICAL DIRECTOR
Darko Aleksic



PROBLEM AUTHORS
Darko Aleksic
Sean McIntyre
John Zhang

SUPPORT
Sonny Chan
Alex Fink
Brett Shikaze

FACULTY SUPPORT
Peter Høyer

RULES & INSTRUCTIONS

The following should be observed for the duration of the contest:

General Conduct

1. You may communicate with your teammates and contest officials only.
2. Use of any electronic device other than your assigned workstation is prohibited. (Devices such as watches are permitted.)
3. You may use as reference any printed materials you brought to the contest.
4. No food or drink is allowed at the workstations!!!

Contest Computer & Resources

1. You may use only the one workstation assigned to you.
2. You may use any tools, including text editors, IDEs, debuggers, and other support applications that are installed on the contest workstation.
3. Access to the internet and network are prohibited except to the official contest page.
4. If you encounter technical difficulties, contact a contest official immediately.

Creating Solutions

1. You may program your solutions in standard C, C++, or Java.
2. Your program should read from standard in and write to standard out.
3. Your program must process the judge input data file in 10 seconds or less.
4. The entire source code for your solution must be in a single source file. If using Java, the main method must be in a class with the specified name (other inner classes are allowed).
5. When you are finished with a program, submit the source code for your solution on the contest page using your contest ID and password.
6. Questions about the problem statements should be sent via the clarifications request form on the contest page.

Enjoy the contest!

CHEAT SHEET

inputFile contents

1 2 3

Java

```
// file: "X.java"; compile: "javac X.java"; run: "java X < inputFile"
import java.io.*;
import java.util.*; // check Java API for useful Scanner methods
public class X {
    public static void main(String[] args) throws IOException {
        BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
        Scanner sc = new Scanner(in.readLine());
        while (sc.hasNext()) System.out.print(sc.next() + " ");
        System.out.println();
    }
} // output: "1 2 3 "
```

C

```
// file: "X.c"; compile: "gcc -Wall X.c"; run: "./a.out < inputFile"
#include <stdio.h>
int main() {
    int x, y, z;
    scanf("%d %d %d", &x, &y, &z);
    printf("z: %d, y: %d, x: %d\n", z, y, x);
    return 0;
} // output: "z: 3, y: 2, x: 1"
```

C++

```
// file: "X.cpp"; compile: "g++ -Wall X.cpp"; run: "./a.out < inputFile"
#include <string>
#include <iostream>
#include <sstream>
using namespace std;
int main() {
    string line; int x, y, z;
    getline(cin, line);
    stringstream in(line);
    in >> x >> y >> z;
    cout << "z: " << z << ", y: " << y << ", x: " << x << endl;
    return 0;
} // output: "z: 3, y: 2, x: 1"
```

Problem A

ALLO, ALLO?

File name: A.{java,c,cpp}

Hansel is organizing a programming contest and would like to contact some potential sponsors. He managed to compile a list of phone numbers for various companies. However, some of the phone numbers contain alphabetic letters (you know how letters on a phone are assigned numerical values).

Help him write a program to convert phone numbers consisting of digits and letters to just digits.



Program Input

The first line will contain the number of test cases n . Each of the following n lines will contain a string consisting of only digits (0-9) and uppercase alphabetic letters ('A'-'Z'). Each string can be up to 15 characters long.

Here is a table of the letter-number correspondence.

2	3	4	5	6	7	8	9
A, B, C	D, E, F	G, H, I	J, K, L	M, N, O	P, Q, R, S	T, U, V	W, X, Y, Z

Program Output

For each test case, output on a single line the numerical representation of the phone number.

Sample Input & Output

INPUT	OUTPUT
2 1866COLT515 555CCPC	18662658515 5552272

Problem B

BANKRUPT BAKER

File name: B.{java,c,cpp}

Wolfgang Puck has an extensive collection of cake recipes. They are separated into different binders depending on the type of cake. Although Wolfgang has restaurant franchises all over the world, he is in a period of hard times and is struggling to afford ingredients for his cakes. What cakes can he create with his small budget?



Program Input

On the first line you are given c ($1 \leq c \leq 100$), the number of binders. Each binder begins with *title*, the name of the binder, then on the next line $m n b$ ($1 \leq m, n \leq 100, 1 \leq b \leq 10^6$) where b is Wolfgang's budget in dollars.

The next m lines are given as *ingredient* c where c ($0 \leq c \leq 5000$) is the price in dollars for one unit of *ingredient*.

Then follow n recipes. Each recipe begins with *name* on a line of its own, then on the very next line k ($1 \leq k \leq 100$). The following k lines are of the form *requirement* x where x is the number of units of the ingredient *requirement* used in the recipe *name*.

For each binder, each *ingredient* will be unique. It is guaranteed that the string *requirement* will be described in the m lines of ingredients. The strings *ingredient* and *requirement* will contain no whitespace.

Program Output

For each binder, output the name of the binder in uppercase letters then on separate lines a list of recipes below Wolfgang's budget in increasing order of cost. If no such recipe exists, print "Too expensive!". If recipes have the same cost put them in alphabetical order. Print a blank line after each binder.

Sample Input & Output

INPUT

```
2
My Favourite Cheesecake
8 3 100
sugar 4
water 0
lemonjuice 3
creamcheese 20
vanilla 5
egg 5
cream 10
strawberry 5
Strawberry Whipped Cream
2
cream 5
strawberry 3
Scrumptious Caramel Topping
3
sugar 6
water 3
lemonjuice 1
Secret Cheesecake Base
5
creamcheese 3
sugar 5
vanilla 1
egg 6
cream 1
Million Dollar Cakes
3 1 999999
costlyflour 500
gold 4500
diamond 5000
Display Cake - Do Not Eat!
3
costlyflour 100
gold 100
diamond 100
```

OUTPUT

```
MY FAVOURITE CHEESECAKE
Scrumptious Caramel Topping
Strawberry Whipped Cream

MILLION DOLLAR CAKES
Too expensive!
```

Problem C

COUNTING CHAOS

File name: C.{java,c,cpp}

Wolfgang Puck's rival, Emeril Lagasse ("BAM!"), recently set the world culinary record in the category of smallest soufflé measuring in at a mere 2 cm! Wolfgang, not to be outdone, decided that he would set a culinary record of his own: the most symmetric marble cake in the world. This is clearly not an easy feat!



As we all know from Wolfgang's bestselling biography, he is a very superstitious chef. In his attempts to create the symmetric cake, he has vowed to remove the cake from the oven only at a palindromic time, i.e., a time that reads the same when read from left-to-right as right-to-left. Not including the current time, when is the next opportunity for Wolfgang to remove his cake?

Program Input

On the first line of the input you are given n , the number of attempts Wolfgang makes to make his symmetric cake. The following n lines contain a string formatted as "HH:MM" indicating the current time on a twenty-four hour clock. (So $0 \leq HH \leq 23$ and $0 \leq MM \leq 59$.)

Program Output

For each attempt, output a string indicating the next palindromic time (not including the current time) on a single line formatted as "hh:mm". When determining if HH:MM is palindromic, ignore all leading zeroes in HH. If HH is zero then ignore all leading zeroes in MM.

Sample Input & Output

INPUT	OUTPUT
3	00:01
00:00	23:32
23:30	15:51
14:59	

Problem D

DELIVERY DEBACLE

Java class name: D. {java,c,cpp}

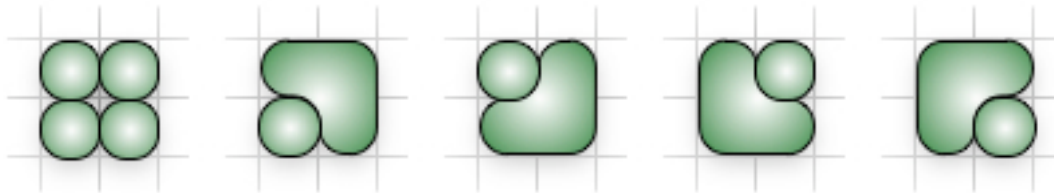
Wolfgang Puck has two very peculiar habits:

- I. He only makes two shapes of cakes. One is square and has an area of one unit. The other is L-shaped and has an area of three units.
- II. He will only deliver cakes packed in very specific box sizes. The boxes are always 2 units wide and are of varying length.



The precise sizes of the cakes Wolfgang makes and a way to pack Wolfgang's cakes in a box of length 6.

One day Wolfgang wondered how many different ways can he pack his cakes into certain sized boxes. Can you help him?



The five ways to pack a box of length 2.

Program Input

The input begins with n , the number of different box lengths. The following n lines contain an integer x ($1 \leq x \leq 40$).

Program Output

For each x output the number of different ways to pack a 2-by- x box with cakes described above. Print a new line after each number. Output is guaranteed to be less than 10^{18} .

Sample Input & Output

INPUT	OUTPUT
2 1 2	1 5

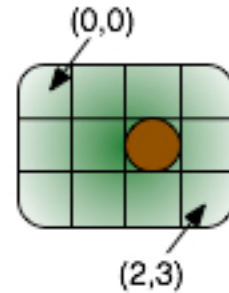
Problem E

EXCLUSIVELY EDIBLE

File name: E.{java,c,cpp}

Hansel and Gretel like cakes, but especially the so called “grid cake” served in Wolfgang Puck’s restaurants. It is made of mn pieces of different cakes, resembling a 2D m -by- n grid when looked at from above (hence the name).

The only thing that Hansel and Gretel do not like about grid cakes is that each of them has to contain a piece of the Scrumptious Caramel Topping cake. It turns out that the only reason Wolfgang Puck has the Scrumptious Caramel Topping cake in his recipe book is because he inherited it from his late great-great-grandmother.

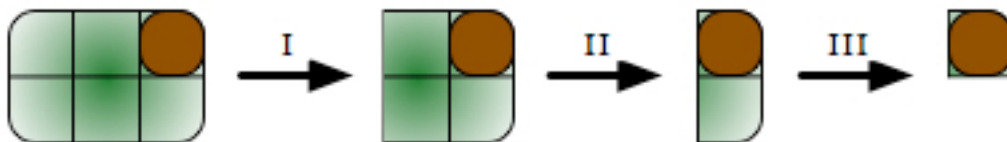


*A three-by-four grid cake
with brown Scrumptious
Caramel Topping cake.*

Neither Hansel nor Gretel want to have the “bad” piece in their portion of the cake, so they came up with the following way to decide who gets the bad piece: first Hansel cuts a piece of the cake along the grid lines, then Gretel does the same and they keep alternating until there is only the Scrumptious Caramel Topping cake piece left.

For example, with a three-by-two grid cake, the illustrations show the following steps:

- I. Hansel cuts the leftmost column. (Gretel is left with a two-by-two grid cake.)
- II. Gretel cuts the leftmost column. (Hansel is left with a one-by-two grid cake.)
- III. Hansel cuts the bottom square off. (Gretel is left with the piece of Scrumptious Caramel Topping cake.)



A sequence of cuts to determine whether Hansel or Gretel gets the bad piece.

Hansel and Gretel have eaten many grid cakes together and have played this game so many times that they know who will take the bad piece before starting. In fact, if they observe Hansel will take the bad piece, then Gretel knows a strategy to ensure Hansel takes the bad piece. Hansel also knows this strategy.

Given the original cake and position of the Scrumptious Caramel Topping cake piece in the grid, who will take the bad piece?

Program Input

The first line of the input file contains a number t ($1 \leq t \leq 50$), the number of test cases. Then t lines follow, each containing $m n r c$ (separated by spaces) where m and n ($2 \leq m, n \leq 20$) are the width and the length of the cake and (r,c) is the zero-based position of the Scrumptious Caramel Topping cake piece in the grid cake ($0 \leq r \leq m-1, 0 \leq c \leq n-1$).

Program Output

For each test case print the name of the person that gets the bad piece assuming that Hansel makes the first cut and that Hansel and Gretel always cut the cake at an optimal location.

Sample Input & Output

INPUT	OUTPUT
2 2 3 0 2 11 11 5 5	Gretel Hansel

Problem F

FLIPPING FRUSTRATION

File name: F.{java,c,cpp}

Wolfgang Puck is not doing so well financially and has accepted a position as a lowly chef at a restaurant franchise named after himself. He is not able to live his normal, lavish lifestyle and as a result he has recently developed some odd tendencies. One tendency in particular is that with his recipe books he can only flip a precise number of pages left (backwards) or a number of pages right (forwards).



Wolfgang must make a dish he knows to be on a certain page of his recipe book. If he starts from the first page, is he able to reach this page? If so, what is the least number of page flips he can make to reach this page?

Program Input

On the first line you are given c ($1 \leq c \leq 100$), the number of occurrences Wolfgang has with his flipping frustration. For each occurrence you are given $n\ l\ r\ t$ ($1 \leq n \leq 1000$, $1 \leq l, r \leq n-1$, $1 \leq t \leq n$) on a line where n is the number of pages in the book, l is the interval left, r is the interval right, and t is the target page number.

Program Output

If it is possible to reach page t from page 1, output on a single line the minimum number of page flips. If it is not possible, print “uh-oh!” on a line.

Sample Input & Output

INPUT	OUTPUT
5	0
10 5 4 1	41
1000 2 1 42	uh-oh!
100 2 4 66	uh-oh!
101 60 70 51	34
100 2 3 98	

Problem G

GOURMET GAMES

File name: G.{java,c,cpp}

Wolfgang Puck is opening a new exclusive restaurant in Las Vegas and is looking for a chef. His nephew Hansel caught him thinking of posting an ad in newspapers. (Note: Wolfgang has heard of “this Internet thingy”, but didn’t find it interesting because one could not spread some Internet on a piece of bread, which is the mere minimum that has to be satisfied for something to be interesting, according to Mr. Puck.)

Hansel knows what the advantages of the Internet job postings are, but he came up with an even better idea – given the success of various reality TV shows and given the fact that the restaurant is opening in Las Vegas, why not organize a sort of a cook-off tournament that will decide who Mr. Puck will hire?

The idea basically combines the Iron Chef™ show with the World Series Of Poker™: Every show m cooks prepare a five-course meal for judges and the best one (according to the said judges) advances to the next round. All candidates are ranked based on the previous experience so, even if the number of the candidates is not divisible by m , some of them can get a “bye” and compete in later rounds. The winner of the final show gets the honour of being the chef in the best Wolfgang Puck’s restaurant ever.



Hansel realized that even with the byes it might not be possible to hold a tournament with m chefs per show, so he is trying to find out what the best m for the given number of candidates is. If you can just help him by letting him know if he can run a tournament as described with n candidates and m chefs per show and, in the case he can, how many shows he needs, he will be so grateful that he might even convince Mr. Puck to share the Secret Cheesecake Base Recipe with you.

Program Input

Input file starts with the line containing a single integer t , the number of test cases. Next t lines each contain two integers separated by spaces, n and m ($2 \leq n \leq 10000$, $2 \leq m \leq n$), where n is the total number of candidates and m is the number of chefs participating in a single TV show.

Program Output

For each test case print the number of shows that need to be run to get the new chef. In the case that the tournament cannot be run with given n and m , print “cannot do this”.

Sample Input & Output

INPUT	OUTPUT
3	2
3 2	cannot do this
4 3	1
4 4	

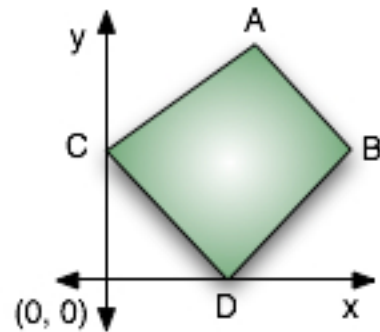
Problem H

HARDLY HARD

File name: H. {java, c, cpp}

You have been given the task of cutting out a quadrilateral slice of cake out of a larger, rectangular cake. You must find the slice with the smallest perimeter that satisfies the following constraints.

If the cake is represented using the first quadrant of the Cartesian plane (yes, it's an infinitely large cake), then your slice is quadrilateral $ABCD$ (see figure). Points A and B are fixed and will be given to you. Also, A, B will lie on a negatively sloping line. Furthermore, points C and D must lie on the positive y -axis and positive x -axis respectively, but it is up to you to determine where these two points should be. A, B, C, D will be distinct points.



Output the minimum perimeter of your slice of cake.

Program Input

On the first line you will be given n ($1 \leq n \leq 100$), the number of test cases. The following n lines each contain $a_x a_y b_x b_y$ ($0 < a_x, a_y, b_x, b_y \leq 10000.0$), the coordinates of points A and B respectively.

Program Output

For each test case, output the perimeter accurate to 3 decimal places on its own line.

Sample Input & Output

INPUT	OUTPUT
1 3.0 1.0 1.0 2.0	7.236