

# Programming Jam 5.0

## problems

March 9, 2019

### Table of Contents

<i>Welcome to Programming Jam 5.0</i> .....	1
<i>Chlorine Compounds</i> .....	2
<i>Calorie Conscious</i> .....	4
<i>Parkour Competition</i> .....	6
<i>Binary Tree Reconstruction</i> .....	9
<i>Teaching Mathematics and English</i> .....	11
<i>Grid Coloring game</i> .....	12
<i>Edge Detection</i> .....	14
<i>Iron Welding Rods</i> .....	17
<i>Do as the romans do!</i> .....	19
<i>Code Validation</i> .....	21
<i>Riyadh Zoo Parrots</i> .....	23
<i>Chain Multiplier</i> .....	26
<i>Shortest Route around the mines</i> .....	27
<i>Nonomino Sudoku</i> .....	29
<i>Organize Storage</i> .....	31
<i>Riyadh Metro</i> .....	33

## Welcome to Programming Jam 5.0

Welcome to Programming Jam 5.0, the fifth year of this competition. The programming Jam competition organization team, the ACM chapter Saudi Arabia and the Prince Sultan University Management are pleased and glad to host this event. On behalf of the team, we appreciate your participation and would like to welcome you and look forward to a successful event.

This time, we will be posting problems in three intervals. At 8am, at 12 pm and 3 pm. You and your team need to solve as many as you can. Your solution will be run against a number of test-cases. If you pass all testcases, you can earn full credit. If some testcases do not pass, you can re-edit and try again until you can get the maximum possible score. You may code in Java, Java8 or C++.

The contest will stop at 5.45 PM Saudi Time.

This year, the winner takes all! Happy hunting for points!!

In this problem we hope that you can say "Hello" to all contestants.

### **Input Format**

None

### **Constraints**

None

### **Output Format**

print "Hello" on screen without quotes! Your line ends with a new line character.

## Chlorine Compounds

A Scientist is conducting an experiment and he wants your help. He wants to know how to generate a chemical solution **Z** with a certain amount of Chlorine.

Unfortunately, he does not have pure Chlorine to make the solution **Z**. Luckily, he has two different solutions that contain some Chlorine: solution **X** and solution **Y**.

Solution **X** contains  $px\%$  Chlorine. Solution **Y** contains  $py\%$  Chlorine.

Help him by writing a program that determines how many Litres of Solution **X** and Solution **Y** is needed to create a Solution **Z** containing  $pz\%$  Chlorine with volume **L** Litres.



### Input Format

Each test case consists of 1 line containing 4 integers ( $px$   $py$   $pz$   $L$ ) representing:

- The percentage ( $px\%$ ) of Chlorine in solution **X**
- The percentage ( $py\%$ ) of Chlorine in solution **Y**
- The percentage ( $pz\%$ ) of Chlorine in the total solution
- The volume **L** of the total solution

These values would be separated by a single space.

### Constraints

$$1 \leq px \leq 100$$

$1 \leq py \leq 100$

$1 \leq pz \leq 100$

$L > 0$

All input and output is Integer. The floats must be rounded off to the nearest integer.

### **Output Format**

Output one line of data containing the amount of Solution X followed by a space and the Solution Y in Liters. Output is also Integer values.

### **Sample Input 0**

30 50 40 10

### **Sample Output 0**

5 5

## Calorie Conscious

Sarah and her friends are opening a pizza pie shop at the campus of a local university. They plan to offer three sizes of pizza slices for the hungry students. The small, medium and large slices have 270, 350 and 450 Calories each. Students can add up to 4 toppings to each pizza pie. Following is a list of toppings available at the pizza shop and the estimated amount of calories per pizza slice.

Topping	Calories		
	Small	Medium	Large
Haloumi	120	135	150
Greenolive	45	48	52
Chicken	95	110	125
Beef	110	125	140
Onion	15	28	45
Pepper	35	45	66
Pineapple	28	35	42
Tomato	36	52	75
Garlic	18	22	26
Feta	125	145	165
Jalapeno	16	20	24
Mushroom	28	33	38
Sausage	95	115	128

You have to write a program that takes an input with different number of combinations, number of slices of pizza and toppings ordered.

### Input Format

The input will contain: An integer  $N$  representing number of combinations (lines). Each line consists of a combination. A combination consists of

- $S$  number of slices followed by
- The size of the slice (Small, Medium or Large) followed by
- A comma separated list of toppings followed by a new line character. Note, there is no spaces between the comma and text before/after.

### Constraints

- $0 < N < 100$
- $0 < S < 100$

There is no spaces between the comma and text before/after. The input ends with a new line character.

### **Output Format**

A Single Integer **V** giving the total value of calories in the order. **V** is 0 for any invalid input

### **Sample Input 0**

```
2
1 Small Tomato
2 Large Pineapple,Chicken
```

### **Sample Output 0**

```
1540
```

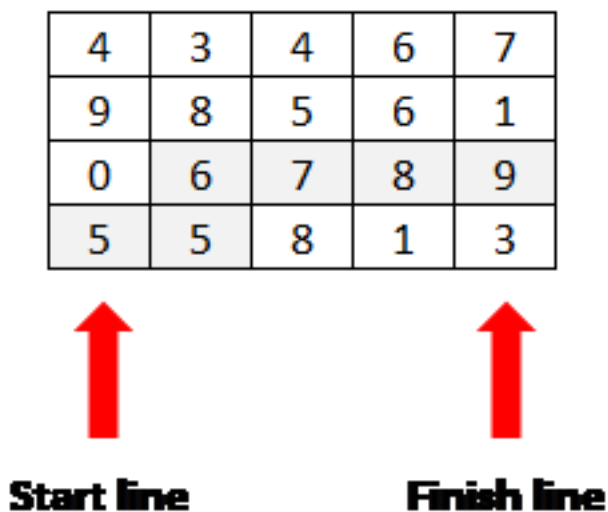
### **Explanation 0**

The total calorie count for this order is 1540

## Parkour Competition

Parkour is the activity of moving rapidly through an area, in a man-made or natural environment, negotiating obstacles by running, jumping and climbing, without the use of any equipment.

The city of Riyadh is organizing a Parkour competition for amateurs with the only rule that a contestant can move to any adjacent position (left or right or up or down but not diagonally) only if the height difference with the current position is at most 1. Given the height map of a parkour event, you are asked to write a program to find the distance of the shortest route between the start line (left edge) and the finish line (right edge). The contestant can start at any position on the start line and being in that position does not require any step. The steps counted are the transitions from one position to the next.



### Input Format

Each test case consists of one line containing 2 integers representing the dimensions  $N$  and  $M$  of the Parkour height map followed by  $N$  lines of  $M$  digits  $D$  representing the heights of the different positions.

### Constraints

$$5 \leq N \leq 50$$

$$5 \leq M \leq 50$$

$$0 \leq D \leq 9$$

## Output Format

For each test case, output one line of data containing the least number of steps to get to the Parkour finish line. If the contestant cannot get across, output the number 0 instead.

### Sample Input 0

```
5 10
2454530236
3451223454
7347354256
9332343218
2632670323
```

### Sample Output 0

```
12
```

### Explanation 0

the shortest path is highlighted, and its length is 12

```
2 4 5 4 5 3 0 2 3 6
3 4 5 1 2 2 3 4 5 4
7 3 4 7 3 5 4 2 5 6
9 3 3 2 3 4 3 2 1 8
2 6 3 2 6 7 0 3 2 3
```

### Sample Input 1

```
7 7
8797784
4876379
2392660
1093459
8722761
5671234
3456746
```

### Sample Output 1

```
17
```

### Sample Input 2

```
5 5
67874
12261
08152
54349
32298
```



**Sample Output 2**

0

**Sample Input 3**

10 8

60717932

26234456

68197014

54248755

95112362

95356477

13518534

53774695

78998716

34551385

**Sample Output 3**

20

## Binary Tree Reconstruction

Consider binary trees whose nodes are labeled with single digits (0 to 9).

Given the preorder (root, left subtree, right subtree) node order and the inorder (left subtree, root, right subtree) node order for the same tree, it is possible to reconstruct the tree, assuming no two nodes in the tree have the same label.

For example, given the preorder traversal order “3 1 0 2 4 6 5” and the inorder traversal order “0 1 2 3 4 5 6” you can compute that the tree these come from is:



You are to write a program to do this reconstruction on any such tree.

### Input Format

The input consists of 2 lines where the first is for the preorder traversal order and the second line is for the inorder traversal order. Each traversal sequence consists of space separated and non-repeated single digits between 0 and 9.

### Constraints

Each traversal line should contain non-repeated single digits between 0 and 9.

### Output Format

For each test case, print the reconstructed tree’s postorder traversal order (left subtree, right subtree, root), using the same input format. All Integer values must be separated by spaces.

### Sample Input 0

```
3 1 0 2 4 6 5
```

0 1 2 3 4 5 6

**Sample Output 0**

0 2 1 5 6 4 3

## Teaching Mathematics and English

One of your friend is teaching Mathematics and English to school children. To make the learning fun, he devised a learning game where he spreads out the letters in a random sequence and then ask the students to identify letters that are placed in correct sequence relative to each other. Of course, there can be more than one sequence, so he asks them to identify the sequence which has the highest number of letters.

To verify the answer, he approached you to develop a program that can provide him the answer automatically once you input the letter sequence.

### **Input Format**

Each test case consists of 1 line of characters. (No spaces) followed by a newline character.

### **Constraints**

A line can have a maximum of 26 characters. All the characters are uppercase letters of the English alphabet.

### **Output Format**

Output is a single Integer value which gives the length of the longest sequence of characters in relative order.

### **Sample Input 0**

```
AFBGCE
```

### **Sample Output 0**

```
4
```

### **Explanation 0**

The sequence ABCE

## Grid Coloring game

The Grid Coloring game consists of coloring a square grid of dimension  $D$ , consisting of white cells and black cells. The game rules are that only the white cells can be colored while the black cells are forbidden (cannot be colored) and the number of cells to be colored on each row and column of the grid is given. You are asked to write a program to find whether it is possible or not to color a given grid.

### Input Format

Each test case consists of

- the dimension  $D$  of the square grid
- $D$  integer numbers  $r_i$  that represent the number of cells to be colored on each row
- $D$  integer numbers  $c_i$  that represent the number of cells to be colored on each column
- $D$  lines of  $D$  characters representing the grid, where 0 represents a white cell and x represents a black cell (forbidden cell that cannot be colored).

### Constraints

$$1 \leq D \leq 50$$

$$0 \leq r_i \leq D$$

$$0 \leq c_i \leq D$$

### Output Format

For each test case, output one line of data containing 1 if the grid can be colored, 0 otherwise.

### Sample Input 0

```
3
2 2 1
3 1 1
000
0x0
000
```

### Sample Output 0

```
1
```

### Explanation 0

The sample input represents the following grid and conditions

	3	1	1
2	Yellow	Yellow	White
2	Yellow	Black	Yellow
1	Yellow	White	White

## Edge Detection

In computer vision, edge detection is an important procedure to find edges in an image. An image is a matrix of  $n$  by  $m$  size, where  $n$  is the number of rows and  $m$  is the number of columns. We will consider only black and white images where black is pixel with value 0 and white is pixel with value 1.

Let us consider the following 8 by 8 image.

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

This image is black so it does not have edges. Let us consider another 8 by 8 black image with white edges.

0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	1	1	1
1	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0
1	0	1	0	1	0	0	0
1	0	0	0	0	1	0	0
1	0	1	1	0	0	1	0

This image is black and 6 edges in total: 3 vertical edges, 2 horizontal edge, and one diagonal edge. In general, an edge is defined as a sequence of 1 values in horizontal, vertical or diagonal direction. An edge is composed of at least of two values of 1.

Write a program that takes as input a matrix, composed on 0 and 1 values, and return as output three numbers, in this order:

1. Number of horizontal edges
2. Number of vertical edges
3. Number of diagonal edges

A diagonal edge can be tilt to the left of the right, in either case it is called diagonal.

### **Input Format**

- Integer n, describes the number of rows to be read in the matrix
- Integer m, describes the number of columns to be read in the matrix
- n number of rows with m number of values, 0 or 1 on each row followed by an nextline character.

### **Constraints**

$$1 < n < 100$$

$$1 < m < 100$$

### **Output Format**

3 comma seperated integer values. Number of horizontal edges, vertical edges and diagonal edges.

### **Sample Input 0**

```
8
8
0,0,0,0,0,0,0,0
1,0,0,0,0,0,0,0
1,0,1,1,1,1,1,1
1,0,1,0,0,0,0,0
0,0,1,0,0,0,0,0
```



```
1,0,1,0,1,0,0,0  
1,0,0,0,0,1,0,0  
1,0,1,1,0,0,1,0
```

### **Sample Output 0**

```
2,3,2
```

### **Explanation 0**

The first input is the number of rows

The second input is the number of columns

Then, every row will have values separated by a comma.

Output: Where 2 is the number of horizontal edges, 3 is the number of vertical edges, and 2 is the number of diagonal edges. The output must be in the same order. There is no space in the output, the three values are just separated by commas.

## Iron Welding Rods

US Iron, a company that specializes in custom hand forged iron, wants to reuse iron rods ends and weld them into one longer rod. Given  $N$  iron rods of different lengths, you are asked to write a program to find the minimum cost to weld all of them into a longer rod, knowing that the cost of welding two iron rods is equal to the sum of their two lengths.

### Input Format

Each test case consists of one line containing one integer  $N$  representing the number of iron rods to weld followed by  $N$  integer numbers  $L$  representing the length of each rod.

### Constraints

$$3 \leq N \leq 100$$

$$1 \leq L \leq 100$$

### Output Format

For each test case, output one line of data containing the minimum cost to weld all the iron rods into a single rod.

### Sample Input 0

```
4 5 4 8 3
```

### Sample Output 0

```
39
```

### Explanation 0

We are given 4 iron rods of lengths 5, 4, 8 and 3. We can weld the rods as follows:

1. Weld the rods with lengths 4 and 3. Now we have three rods with lengths 7, 5 and 8.
2. Weld the rods with lengths 7 and 5. Now we have two rods with lengths 12 and 8.
3. Finally weld these two rods.

The total cost for welding the 4 rods is  $20 + 12 + 7 = 39$ . This is the minimum cost for welding the 4 rods into one.

Other ways of welding the rods would yield either the same cost or a higher one. For example:

1. Weld the rods with lengths 5 and 4. Now we have three rods with lengths 9, 8 and 3.
2. Weld the rods with lengths 9 and 8. Now we have two rods with lengths 17 and 3.

3. Finally, weld these two rods.

The total cost for welding the 4 rods would be  $9 + 17 + 20 = 46$

**Sample Input 1**

5 21 29 18 33 9

**Sample Output 1**

247

**Sample Input 2**

6 63 11 42 21 35 53

**Sample Output 2**

549

**Sample Input 3**

7 2 11 15 20 7 17 6

**Sample Output 3**

205

**Sample Input 4**

10 56 15 35 86 23 48 52 25 65 74

**Sample Output 4**

1535

## Do as the romans do!

The Roman numeral system represents numbers as a string of letters from the Latin alphabet, according to the table below. The Roman symbols in a string always appear in descending value from left to right. The value of each symbol is added to obtain the value of the string.

Example: 2019 is written MMXIX 3993 is written MMMCMXCIII

Write a program that takes a Roman numbers as a String converts it to integer value and decrements it by 1. The resulting number is printed as Roman numerals (String) with upper-case letters.

<b>Symbol</b>	<b>Decimal value</b>
I	1
IV	4
V	5
IX	9
X	10
XL	40
L	50
XC	90
C	100
CD	400
D	500
CM	900
M	1000

### **Input Format**

Each test case consists of a String representing a Roman number greater than 0.

### **Constraints**

The input string should only contain the symbols represented in the table above, and have a length  $\geq 1$ .

### **Output Format**

For each test case, output the Roman number as a String representing the input Roman number decremented by 1.

For invalid input, the program prints 0.

### **Sample Input 0**

```
MCM
```

### **Sample Output 0**

```
MCM
```

## Code Validation

A Markup Language (ML) is a computer language that uses tags to define elements within a document. The following example shows content of ML document:

```
<table>
<row>
  <cell>fruit</cell>
</row>
<row>
  <cell>vegetable</cell>
  <cell>potato</cell>
  <cell>onion</cell>
  <cell>carrot</cell>
</row>
<row>
  <cell>cars</cell>
  <cell>toyo</cell>
  <cell>ford</cell>
</row>
<row>
</row>
</table>
```

We say that this document is well-formed since:

1. There is only one root element (table)
2. Every element begins with a starting tag (<table>) and ends with an ending tag (</table>)
3. Every (parent) element may have zero or more children elements of a single type (row element may have only cell elements)

We say this document is valid since it respects a specific dictionary that we described in this example as “table row cell” where this means:

1. **table** is the root element
2. table may have only children of type **row**
3. row may have only children of type **cell**
4. cell cannot have children

You are asked to write a program to check if the document is well-formed and valid by considering a predefined dictionary.

## Input Format

A string containing the input values.

## Constraints

The maximum length of the input String is 8192 characters.

## Output Format

Integer value 0 or 1

0 means if the input is invalid

1 means if the input is valid.

## Sample Input 0

```
<table> <row> <cell>vegetable</cell><cell>potato</cell><cell>onion</cell><cell>carrot</cell>
</row> </table>
```

## Sample Output 0

1

## Explanation 0

、

vegetable potato onion carrot

、

## Riyadh Zoo Parrots

Parrots are beautiful creatures, some of their types can learn tricks and language. The word set for parrots is limited so they end up using same words for couple of things or you have to teach them intelligently. There are currently number of imported parrots in Riyadh Zoo only from four descents (Africa, Chile, Saudia, Norway). Inspired on your latest visit to Zoo You also bought a new parrot who can learn X set of alphabets.

Being a computer Scientist you installed a brain enhancer chip to parrot to make him intelligent. This has enabled your parrot to learn words as well as distinguish wrong words. Currently only intelligence he got is that he can distinguish whether he can learn a word from list of words or not.

The parrot can learn the words according to his descent and the date of birth. If the parrot is from Africa he can learn all words that starts or contains A. The parrots can learn each day one word from the day he was born. But can learn only one size words such as their age in days compared (finding remainder) to length of country name. \*Suppose the parrot is born on 26th December 2016 and Today is 9th March 2019, then total days are 803 while number of letters are 6, so we need to find 5 letter words while rest cannot be learned. \*

Given a list of words of different size, your job is to find out how many words your AI parrot can learn and how many he cannot of the same size.

### Input Format

```
Africa 26/12/2016
AA
AB
ATILT
ATIMY
ATLAS
0
```

First line represents the Descent and Date (DD/MM/YYYY). Each subsequent line contains different dictionary words followed by a line containing 0 as a sentinel character.

### Constraints

The letters that can be learned are already given. The acceptable Descent of a parrot are



Africa  
Chile  
Saudia  
Norway

Any other parrot bought will Learn only one word PJAM

### Output Format

```
3 2
ATILT
ATIMY
ATLAS
```

First line shows that 3 letters are learned and 2 are not learned other lines shows the letter learned

### Sample Input 0

```
Africa 26/12/2016
AA
AB
ATILT
ATIMY
ATLAS
0
```

### Sample Output 0

```
3 2
ATILT
ATIMY
ATLAS
```

### Sample Input 1

```
Oman 26/12/2016
AA
AB
AZ
ATILT
ATIMY
ATLAS
ATTAP
ATTAR
ATTIC
0
```

### Sample Output 1

```
PJAM
```

### Sample Input 2

```
Africa 5/1/2019
```

AA  
AB  
AZ  
ATILT  
ATIMY  
ATLAS  
ATTAP  
ATTAR  
ATTIC  
0

### Sample Output 2

0 9  
None

## Chain Multiplier

One of the biggest problems in matrix chain multiplication is finding a way to parenthesize i.e. when multiplying three matrices: A B C, one can multiply A with B first or B with C. e.g.

$A * (B * C)$  or  $(A * B) * C$

In both cases, the answer would be the same since Matrix multiplication is associative. However, the number of required operations would be different.

Write a program that determines the order which gives the least number of multiplication operations.

### Input Format

The input file should contain the dimensions of the matrix each on a separate line.

### Constraints

The input must be positive integers.

### Output Format

One Integer giving the minimum number of multiplication operations needed. -1 is printed if multiplications is not possible.

### Sample Input 0

```
10
5
8
4
10
```

### Sample Output 0

```
760
```

### Explanation 0

The above input and output samples correspond to the multiplication of the following four matrices:

A (10\*5) B (5\*8) C (8\*4) D (4\*10)

## Shortest Route around the mines

Ali lives in a city ravaged by war. All the city has become a minefield. Every day he has to go from his home (H) to school (S). Given a matrix representing his neighborhood, you are asked to write a program to find the shortest integer distance between home and school avoiding the mines that could be in the way, knowing that there will always be a valid route from home (H) to school (S) and that the valid moves are into any adjacent or diagonal safe space.

### Input Format

Each test case consists of one line containing 2 integers representing the dimensions N and M of the matrix followed by N lines of M characters. The characters used are as follows:

.	Safe space
#	Mine
H	Home
S	School

### Constraints

5 <= N <= 100  
5 <= M <= 100

### Output Format

For each test case, output the shortest integer distance between Ali's home (H) and school (S) without stepping on a mine.

### Sample Input 0

```
8 10
.....
...#....
...#....
H...#..S..
...#....
...#....
.....
.....
```

### Sample Output 0

7

### Sample Input 1

6 8

```
....H#.  
.....#.  
.....#S  
.....#.  
...####.  
..#.....
```

### Sample Output 1

11

### Sample Input 2

10 10

```
..#.....  
..#.....  
.....#.  
.....#.  
H..####S  
..#.....  
..#.....  
..###....  
....###..  
.....
```

### Sample Output 2

9

### Sample Input 3

13 5

```
.....  
#.S.#  
.###.  
...#  
...#.  
.###.  
#...#  
.#.##  
...#  
.###.  
.##.#  
.###.  
#..H.
```

### Sample Output 3

12

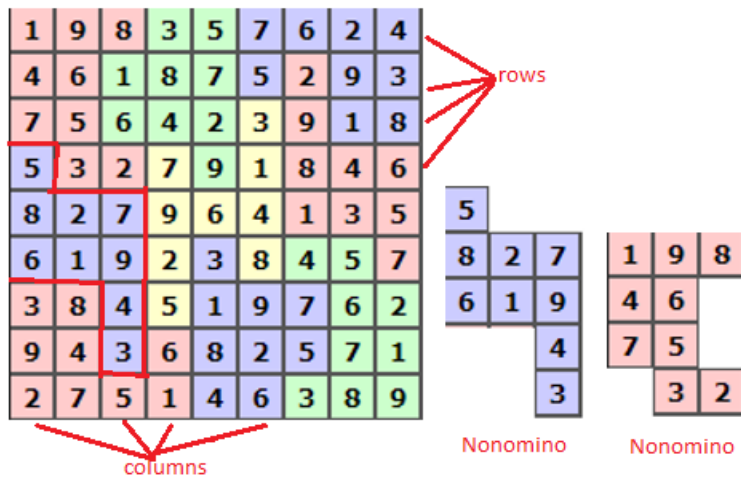
## Nonomino Sudoku

Sudoku is a logic based combinatorial number placement puzzle. The objective is to fill a 9×9 grid with digits so that each column, each row, and each of the nine 3×3 subgrids that compose the grid (also called "boxes", "blocks", or "regions") contain all of the digits from 1 to 9. The puzzle setter provides a partially completed grid, which for a well-posed puzzle has a single solution.

A variation on the classical Sudoku is the Nonomino Sudoku (also known as jigsaw Sudoku). The puzzle is divided into 9 regions (or nonomino), each containing 9 cells. Each cell contains Integer values from 1 to 9. As standard Sudoku rules apply, each region must contain a uniquely placed integer from 1 to 9. A region must not have multiple instances of the same integer.

For this problem, we will consider the H-shaped nonomino Sudoku puzzle. Each region or nonomino contains 9 cells. The correct solution will have

1. All integers 1 to 9 appearing exactly once in a nonomino
2. All integers 1 to 9 appearing exactly once in each row
3. All integers 1 to 9 appearing exactly once in each column



Write a program that reads the nonomino Sudoku puzzle solutions and verifies if the solution is correct.

### Input Format

A matrix of 9 rows and columns of Integer values (1 to 9) separated by spaces.

### Constraints

All inputs are integer values  $i$  such that  $0 \leq i \leq 9$ .

### Output Format

Program outputs a single integer 0 or 1 to indicate valid or invalid solution. 1 if valid 0 if invalid

### Sample Input 0

```
1 9 8 3 5 7 6 2 4  
4 6 1 8 7 5 2 9 3  
7 5 6 4 2 3 9 1 8  
5 3 2 7 9 1 8 4 6  
8 2 7 9 6 4 1 3 5  
6 1 9 2 3 8 4 5 7  
3 8 4 5 1 9 7 6 2  
9 4 3 6 8 2 5 7 1  
2 7 5 1 4 6 3 8 9
```

### Sample Output 0

```
1
```

### Explanation 0

All rows have unique values 1-9 All columns have unique values 1-9 All nonominos have unique values 1-9

## Organize Storage

A store owner wants to organize all the items in his store inside boxes in the most possible optimal way. The question is how many boxes are required to hold the items? There are up to ten 5x5 boxes available. Items of different shapes will be given as a set of graphical input: The Hash sign (#) represents a solid object while the period (.) represents a free space. Items can be rotated to fit inside a box; free space can overlap and go outside of box boundaries. Each item will have at least one solid item unit (#) in it.

Some of the items can be “hollow”; and smaller items can fit inside the free space. Note that the rotated pieces must remain in the same relative positions of each other. Rotations can be of 90 degrees only, so each item will have at most 4 unique shapes (because of possibility of rotating the item by 90 degree four times); Check input/output samples for clarifications.

### Input Format

The first line in the input is an integer ( $1 \leq n < 30$ ) that represents the number of items to be stored in the boxes. A graphical representation of the items is presented by the (#) and (.) symbols. Items are separated by a blank line in between them.

### Constraints

Only two-dimensional space will be considered for this problem Items are not larger than 5x5  $1 \leq n < 30$ , where n is the number of items to be stored

### Output Format

A positive integer that represents the minimum number of boxes required to hold the items.

### Sample Input 0

```
4
#####
#...#
#...#
#...#
#####

##
```



```
##
```

```
###.
```

```
..#.
```

```
..#.
```

```
#####
```

```
#####
```

```
#####
```

```
#####
```

```
#####
```

### Sample Output 0

```
2
```

### Sample Input 1

```
2
```

```
.####
```

```
#####
```

```
##...
```

```
##...
```

```
##...
```

```
###..
```

```
###..
```

```
###..
```

```
....
```

```
...#
```

### Sample Output 1

```
1
```

## Riyadh Metro

The Riyadh Metro is a rapid transit system under construction in the city of Riyadh, the capital of Saudi Arabia. It will consist of six metro lines spanning a total length of 176 kilometers, with 85 stations. For the trial phase the Metro has acquired Six passenger carts, where each passenger cart can accommodate 50 passengers. In order to save energy, the number of carts used are exactly that can accommodate people.



During the trail phase, Metro has decided to award all passengers unique gifts. But before giving the gifts the gifts are shuffled and then distributed.

You are given a task by your manager to find out the total number that is possible for distribution of gifts. Hint: how many shuffles can be made.

In order to find the number of passenger carts, the number of cars, buses or SUV arrived are metro station are used. Each vehicle is filled to its capacity. The capacity of car is 4, SUV 7 and bus is 25

### **Input Format**

There first line explains how many inputs are coming. Rest of lines are vehicle type separated by no of vehicles that have arrived. In the first input it is shown as following line shows the input

```
2  
CAR 2  
SUV 2
```

Here no of inputs that coming are 2 while the vehicles that are arrived

- 2 cars and
- 2 SUV

### Constraints

1. The number of carts can be  $1 < X < 6$ . while the number of passengers cannot be more than the capacity of total carts.
2. Even for a single passenger you will use one full cart.
3. If the number of passengers are more than the total capacity A message is sent that "The passengers cannot go in one go"

### Output Format

- There are two lines of output;
- first line prints the number of carts needed;
- second line prints the possible number of gift combinations that can be given.

as for above input the following explanation is given.

You need 1 Cart to Transfer

The gifts can be given in 1124000727777607680000

### Sample Input 0

2

CAR 2

SUV 2

### Sample Output 0

You need 1 Cart to Transfer

The gifts can be given in 1124000727777607680000 ways

### Sample Input 1

4

BUS 16

SUV 34

CAR 23

### Sample Output 1

You cannot go in one go