

## Problem A. Very Wealthy Mole

Input file:            *standard input*  
Output file:           *standard output*  
Time limit:            2 seconds  
Memory limit:         64 MiB

- *What would you like to do in the meantime, wealthy moles?*
- *What if we make some calculations?*
- *That's a good idea!*

From the Soviet animated film *Thumbelina*

One very wealthy mole knows only nonnegative integers. He can perform three operations on his abacus:

- (1) obtain the number  $2x$  from a number  $x$ ;
- (2) obtain the number  $2x+1$  from a number  $x$ ;
- (3) obtain the number  $\lfloor x/2 \rfloor$  (the integer part of  $x$  divided by 2).

Each morning the mole chooses a pair of integers  $x$  and  $y$  such that  $1 \leq x < y \leq 2^l - 1$  and, during the day, obtains the number  $y$  from the number  $x$  by performing a sequence of operations on his abacus. The mole is good at arithmetics, so he always uses the shortest sequence of operations sufficient for obtaining the number  $y$  from the number  $x$ .

How many abacus operations will the mole perform in total until he exhausts all pairs of integers in the range from 1 to  $2^l - 1$ ?

### Input

The only line contains the integer  $l$  ( $2 \leq l \leq 10^{18}$ ).

### Output

Find the number of abacus operations that the mole will perform and output this number modulo  $10^9 + 7$ .

### Example

<i>standard input</i>	<i>standard output</i>
2	4

On the first day the mole obtains the number 2 from the number 1 by performing the first operation. On the second day the mole obtains the number 3 from the number 1 by performing the second operation. On the third day he obtains the number 3 from the number 2 by performing the third and then the second operation.

## Problem B. Peaceful Atom (Division 1 Only!)

Input file: *standard input*  
Output file: *standard output*  
Time limit: 4 seconds  
Memory limit: 64 MiB

*The most frightening word in nuclear physics is “Oops!..”*

The Beloyarsk Nuclear Power Plant has been severely damaged in the recent earthquake of magnitude 7 that has stricken the Ural mountains. The nuclear reactor is intact, but almost all control systems are not functioning. In particular, the sensors are out of order and the staff can only guess what is happening inside the reactor. The reactor must be stopped as soon as possible! For this, the staff has to know the current positions of the control rods.

Control rods are used for controlling the nuclear reaction. Their steel shells are filled with boron, which is capable of absorbing a large number of neutrons. When the rods are inserted into the reactor core, they absorb neutrons and the reaction slows down. Conversely, when the rods are removed from the core, the number of neutrons increases and the reaction accelerates. At the BNPP, the rods can be shifted from the zero position, when they are removed completely, to the depth of  $n$  mm, which is the complete insertion. Inserting the rods without knowing the state of the reactor may be dangerous. For example, when control rods were inserted into the reactor core at the Chernobyl Nuclear Power Plant in 1986, the reactor went out of control. That is why it is important to determine the current positions of the rods before taking any further actions.

The staff have a log with all the changes in the positions of the rods. Unfortunately, the log contains only the numbers of millimeters by which the rods were shifted, but it does not say in which direction—up or down. Now the staff need your help.

### Input

The first line contains integers  $n$ ,  $s$ , and  $k$ , which are the maximum insertion depth of the rods, the initial insertion depth, and the number of entries in the log, respectively ( $0 \leq s \leq n \leq 10^9$ ;  $n > 0$ ;  $1 \leq k \leq 40$ ). In the second line you are given  $k$  positive integers, which are the values of the shifts of the rods in chronological order. It is guaranteed that there exists a sequence of shifts consistent with the log such that during this sequence of shifts the rods are always inserted at least 0 mm and at most  $n$  mm deep.

### Output

Output the minimum and maximum possible insertion depths of the rods after all the shifts.

### Example

<i>standard input</i>	<i>standard output</i>
10 3 3 4 5 1	1 3

## Problem C. Get-Together at Den's

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 64 MiB

*Advertising for beer and beer-based beverages must not use images of people and animals.*

Russia's Federal Law "On Advertising", Ch. 3, Art. 22

Den invited  $n$  friends to his home. The friends were glad but first went to Auchan to buy some juice. The  $i$ th friend bought  $a_i$  liters of juice. Then the friends came to Den and sat down to drink the juice. Everybody, including Den, drank the same amount of juice.

When there was no more juice, Misha started complaining about having bought more juice than he had drunk. Den didn't want Misha to get upset, so he took one hundred roubles out of his pocket and gave the money to Misha to compensate him for the juice he had bought but hadn't drunk himself. Then some more of Den's friends started complaining about the same issue. Den had no more money, so he suggested that all the friends who had bought more juice than they had drunk themselves should fairly divide his one hundred roubles between them. How should Den's friends divide the money? Assume that Den paid not only for the juice he had drunk but also for all the juice that had been drunk by the people who hadn't bought it.

### Input

The first line contains the number  $n$  of Den's friends ( $2 \leq n \leq 100$ ). In the second line you are given the integers  $a_1, \dots, a_n$  ( $0 \leq a_i \leq 100$ ). The sum of all  $a_i$  is positive.

### Output

Output how much of the one hundred roubles the  $i$ th friend should take. Round the amounts down to an integer number of roubles.

### Examples

<i>standard input</i>	<i>standard output</i>
3 10 10 10	33 33 33
2 10 0	100 0

## Problem D. The Island of Märket (Division 1 Only!)

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 64 MiB

Märket is uninhabited island in the Baltic Sea. Historically, the western part of the island belongs to Sweden, while the eastern part belongs to Finland. In 1885, Finns built a lighthouse on the island. There were no clear maps of the island at that time, so later Sweden declared that the lighthouse had been built on its territory. To settle the territorial dispute, Sweden and Finland marked the border on the island so that the areas of the Swedish and Finnish parts were equal and the lighthouse was completely on the Finnish territory.

We assume that Märket is a rectangle on the plane and the lighthouse is a square located strictly inside this rectangle. The sides of the rectangle and of the square are parallel to the coordinate axes. The southwest corner of the island has coordinates  $(0,0)$  and the northeast corner has coordinates  $(w,h)$ . The border must be a polyline with vertices at points with integer coordinates. The line must connect the north coast with the south coast and its segments must be parallel to the coordinate axes. The line must not have common points with the west or east coasts of the island and must have exactly one common point with the north coast and one common point with the south coast. The border must not cross the lighthouse but can touch it.

Your task is to draw a border between the countries on the island map.

### Input

The first line contains the integers  $w$  and  $h$ , which are the dimensions of the island ( $3 \leq w, h \leq 10^6$ ;  $w \cdot h$  is even). In the second line you are given integers  $x$ ,  $y$ , and  $d$ , which are the coordinates of the southwest corner of the lighthouse and the length of a side of the lighthouse ( $d > 0$ ;  $0 < x < w-d$ ;  $0 < y < h-d$ ).

### Output

Output the number of vertices of the polyline in the first line. In the following lines output the integer coordinates of the vertices starting from the vertex on the north coast and ending with the vertex on the south coast. The vertical and horizontal segments of the line must alternate; the first and the last segments must be vertical. The lengths of all the segments must be positive. The polyline must not intersect or touch itself. The number of vertices must not exceed 1 000. If there are several ways to draw the border, output any of them. If it is impossible to draw a border satisfying the above conditions, output “Impossible”.

### Examples

<i>standard input</i>	<i>standard output</i>
6 3 1 1 1	6 4 3 4 2 1 2 1 1 4 1 4 0
10 10 1 1 8	Impossible

## Problem E. Poetic Foot

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 64 MiB

*And telling trochee from iambic  
Was quite beyond his wit, we found.  
A. S. Pushkin*

A poetic *foot* is a sequence of syllables one of which is stressed. The type of foot is determined by the number of syllables in it and by the position of the stressed syllable. A disyllabic foot is called a *trochee* if the first syllable is stressed and an *iamb* if the second syllable is stressed. A trisyllabic foot is called a *dactyl*, *amphibrach*, or *anapaest* if the first, second, or third syllable is stressed, respectively.

Usually, each line of a poem consists of several feet, but the last foot in a line may be shortened to the first one or two syllables. All the feet in a poem must be of the same type.

School students are often asked to determine the meter (or a rhythmical structure) of a poem in their literature classes. The first step in finding the meter is to determine the type of foot used in the poem. You can help students by automating this process.

### Input

The first line contains the number of lines in a poem  $n$  ( $2 \leq n \leq 100$ ). The poem is given in the following  $n$  lines. Each word of the poem consists of one or more syllables. If there are several syllables in a word, they are separated by a hyphen. Each stressed syllable is enclosed in square brackets. It is guaranteed that there are at least four syllables in each line of the poem. We have deleted all punctuation marks from the text, and now it contains only uppercase and lowercase English letters, spaces, hyphens, and square brackets. The length of each line is at most 50 symbols.

### Output

Output the type of foot used in the poem: “trochee”, “iamb”, “dactyl”, “amphibrach”, or “anapaest”. If the type of foot cannot be determined, output “not a poem”.

### Example

<i>standard input</i>
4 My [un]-cle [high] i-[deals] ins-[pire] him But [when] past [jo]-king [he] fell [sick] he [real]-ly [forced] one [to] ad-[mire] him and [ne]-ver [played] a [shrew]-der [trick]
<i>standard output</i>
iamb

## Problem F. Nanotechnologies

Input file:            *standard input*  
Output file:           *standard output*  
Time limit:            2 seconds  
Memory limit:         64 MiB

Specialists at the Nanotechnology Lab at the Ural Federal University work on developing a computer that would operate at the level of individual atoms. They have recently learned that  $n$  tantalum atoms arranged on a plane in a special way can be used for constructing a quantum computer with earlier unattainable characteristics.

Unfortunately, the theory describes only some of the physical characteristics of such a system of atoms. The physicists must find an arrangement of atoms that would provide the necessary effect. The scanning tunneling microscope used in the project can place tantalum atoms only at points whose coordinates are integers with absolute values not exceeding  $10^6$ ; the unit of the coordinate system is one nanometer.

The scientists ask you to help them determine the coordinates of the atoms. They have eliminated the unnecessary physical information, providing you only with the squared distances between all the pairs of atoms.

### Input

The first line contains the total number of atoms  $n$  ( $2 \leq n \leq 100$ ). In the following  $n$  lines you are given an  $n \times n$  matrix  $a_{ij}$  with integer elements. The number  $a_{ij}$  is the required squared distance between atoms numbered  $i$  and  $j$  ( $0 \leq a_{ij} \leq 10^9$ ;  $a_{ij} = a_{ji}$ ;  $a_{ii} = 0$ ).

### Output

If there is an arrangement satisfying the given matrix of squared distances, output  $n$  lines. The  $i$ th line must contain the coordinates of the  $i$ th atom; they must be integers with absolute values not exceeding  $10^6$ . If there are several possible answers, output any of them. If it is impossible to arrange the atoms as required, output "Impossible".

### Examples

<i>standard input</i>	<i>standard output</i>
3 0 1 4 1 0 1 4 1 0	0 0 1 0 2 0
3 0 1 5 1 0 1 5 1 0	Impossible
3 0 1 0 1 0 1 0 1 0	5 -5 5 -4 5 -5

## Problem G. Prediction Contest

Input file:            *standard input*  
Output file:           *standard output*  
Time limit:            2 seconds  
Memory limit:         64 MiB

At the ICPC World Finals, four sets of gold medals, four sets of silver medals, and four sets of bronze medals are awarded; thus, top twelve teams win medals.

There were few days left before the World Finals in Orlando. Veterans of collegiate programming contests from Ural State University were hotly debating chances of the teams for medals at the forthcoming contest. Vadik suggested that every person who wanted to make a prediction for the results of the finals should pool five dollars and state several guesses of the form “The team of university  $X$  will win a medal of type  $Y$ .” The veterans who made the maximum number of true guesses would share the pooled money between them.

But the veterans refused saying that the results of the finals are always very difficult to predict and nobody wanted to lose five dollars. Then Vadik changed the conditions of the prediction contest. He said that the participants didn’t have to pay, but everybody who made the maximum number of true predictions would receive five dollars from Vadik. The veterans agreed to these conditions, and each of them stated several predictions about which team would win which medal.

Given the result of the World Finals and all the predictions made, find the amount of money that Vadik had to pay to the winners of the competition.

### Input

In the first twelve lines you are given the names of the universities winning the medals of the World Finals. The universities are listed in the order from the first place to the twelfth place.

In the following line you are given the number  $n$  of participants of the prediction contest ( $1 \leq n \leq 100$ ). Then go  $n$  blocks containing the veterans’ predictions. The  $i$ th block starts with the number of predictions  $k_i$  made by the  $i$ th participant. The predictions are given in the following  $k_i$  lines, each prediction has the form `<name_of_university> : <medal>`. Here, `<medal>` is the word “gold”, “silver”, or “bronze”. The predictions of each participant are consistent: no university can win more than one medal and the total number of medals of each kind is at most four.

The names of the universities consist of uppercase and lowercase English letters, underscore characters, dots, and commas. The length of each name is in the range from one to thirty symbols.

### Output

Output the amount of money that Vadik paid according to the result of the prediction contest. If more than one participant made the maximum number of true predictions, Vadik paid five dollars to each of them.

## Example

<i>standard input</i>	<i>standard output</i>
Zhejiang_U U_of_Michigan_at_Ann_Arbor Tsinghua_U St._Petersburg_SU Nizhny_Novgorod_SU Saratov_SU Friedrich_Alexander_U Donetsk_National_U Jagiellonian_U_in_Krakow Moscow_SU Ural_SU U_of_Waterloo 3 6 Moscow_SU : gold St._Petersburg_SU_of_ITMO : gold Warsaw_U : gold Tsinghua_U : gold Nizhny_Novgorod_SU : silver Saratov_SU : silver 6 Warsaw_U : gold Saratov_SU : gold Tsinghua_U : gold Donetsk_National_U : silver St._Petersburg_SU_of_ITMO : silver Ural_SU : bronze 6 Zhejiang_U : gold Tsinghua_U : gold Shanghai_Jiao_Tong_U : gold Fudan_U : gold Moscow_SU : silver Nizhny_Novgorod_SU : silver	15

## Problem H. New Year Cruise

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 64 MiB

*And I drank it straight down.*

In the life of Russian people there is always a place for a holiday! Even more so when several causes for celebration follow each other — New Year, Orthodox Christmas, Old New Year. . . People celebrate these holidays in different ways. Some of them like to be together with their family and friends, while others prefer to go for a travel. The company “Siberian Railroads” decided to combine these two variants by introducing a special cruise train “Booze” traveling along the route Vladivostok–Moscow–Vladivostok. The train would cruise once in a year only, during the New Year holidays. It would be adapted for celebrating each of the holidays up until the following holiday or until arriving to the destination.

When the company started selling tickets for the train, they turned out to be in great demand. The sale was stopped on the third day because a train for all those wishing to go on the cruise would be too long to fit any station. The company’s managers calculated the number of tickets sold for the travel between each pair of stations, and they wanted to know the number of cars in the train sufficient to accommodate all the passengers.

### Input

The first line contains the number  $n$  of stations where the train stops when traveling in one direction ( $2 \leq n \leq 100$ ). The stations are numbered from 1 to  $n$  in the order the train passes them. Vladivostok is numbered 1 and Moscow is numbered  $n$ . In the following  $n$  lines you are given an  $n \times n$  matrix with integer elements  $a_{ij}$ . The element  $a_{ij}$  is the number of people who have bought a ticket from station  $i$  to station  $j$  ( $0 \leq a_{ij} \leq 1000$ ;  $a_{ii} = 0$ ); these people will board the train traveling from Vladivostok to Moscow if  $i < j$  and the train traveling in the opposite direction if  $i > j$ .

### Output

Output the minimal number of cars in the train that is sufficient to accommodate all the passengers. There are exactly 36 seats in each car.

### Example

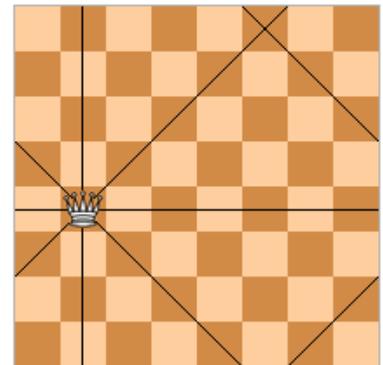
<i>standard input</i>	<i>standard output</i>
3 0 180 180 0 0 180 360 0 0	10

## Problem I. Zinium 2

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 64 MiB

As you remember, Zinium is a chessboard of size  $n \times n$  cells. The cell in its bottom left corner has coordinates  $(1, 1)$ , and the cell in its top right corner has coordinates  $(n, n)$ . According to a legend, if  $n$  queens are placed on the board so that none of them attacks another, the released energy of Zinium will change the world beyond recognition. Rivers will turn back, the sky will fall onto the earth, people will learn to call things by their proper names...

Everybody had believed the legend about Zinium until Igor got hold of the artifact. He managed to place  $n$  queens as required but nothing happened (at least he didn't notice anything unusual). Then Igor conjectured that the chessboard in the legend was not usual but toric. A toric  $n \times n$  chessboard is obtained from the usual chessboard of the same size by gluing together its upper and lower ranks as well as its leftmost and rightmost files. The figure shows the cells of a toric  $8 \times 8$  chessboard attacked by a queen. Igor wants to check his conjecture by placing  $n$  queens on the toric board so that none of them attacks another. Help him to do this.



### Input

The only line contains the integer  $n$  ( $4 \leq n \leq 10^5$ ).

### Output

If the queens can be arranged as required, output "Yes" in the first line and  $n$  integers in the second line. The  $i$ th number must be the  $y$  coordinate of the queen whose  $x$  coordinate is  $i$ . If there are several possible arrangements, output any of them. If the queens can't be arranged as required, output "No" in the only line.

### Examples

<i>standard input</i>	<i>standard output</i>
5	Yes 2 4 1 3 5
8	No

## Problem J. Seismic Waves

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 64 MiB

*In case of fire, exit building before  
tweeting about it.*

Nobody is surprised nowadays hearing a story about someone writing to a blog or a social network about an accident before calling the rescue service. Knowing this attachment to internet technologies, the Japanese authorities suggest people to use Twitter to alert their friends and relatives about an earthquake. Indeed, since the propagation velocity of seismic waves is no more than five kilometers per second, people who are far enough from the epicenter will be able to read about the earthquake several seconds before the wave hits them. These seconds may become crucial for some of them. . .

Each Twitter user has a list of *followers*. A tweet (message) of a user is seen by all of his followers. Each of them can *retweet* this tweet. As a result, it will become available to all their followers. These followers can in turn retweet the retweet, and so on. . . Retweet gets the string “RT @*nick*: ” appended to its beginning, where *nick* is the name of the user whose tweet (or retweet) is retweeted. Since the length of a tweet must not exceed 140 symbols, any retweet violating this restriction will not be sent.

The Japanese Fire and Disaster Management Agency wants to test the idea of Twitter alerts by conducting an experiment. They have chosen a group of people and want to calculate how many of them will learn about an earthquake, if only person will be notified of it.

### Input

The first line contains the number  $n$  of Twitter users chosen for the experiment ( $2 \leq n \leq 100$ ). The users are described in the next  $n$  lines in the following format: username, the number of followers, and the list of their names. The lists of followers can only contain names of other users chosen for the experiment. The names are distinct nonempty strings of length at most 30, consisting of uppercase and lowercase English letters. The last line contains a tweet about an earthquake sent by the user whose name is the first in the input. The tweet is a nonempty string consisting of symbols with ASCII codes in the range from 32 to 127; **spaces at end of the tweet are allowed**; the length of the string is at most 140 symbols.

### Output

In the first line output the number of users who will learn about the earthquake. Then output their names in any order, one name per line.

### Example

<i>standard input</i>	<i>standard output</i>
5 Miyako 1 Sae Yuno 2 Miyako Sae Nazuna 1 Nori Sae 1 Yuno Nori 1 Nazuna Hidamari is going to collapse!!!	3 Miyako Sae Yuno

## Problem K. Spacious Office

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 64 MiB

Programmers at the company “Perimeter” are working on  $n$  software projects. Their boss Shiftman understands the importance of comfortable working conditions. There is neither dress code nor fixed work schedule in the company, but there always is tea and fresh kiwis in the kitchen. When the team of the “Diplodocus” project complained that their room was too crowded after new employees had joined the company, Shiftman understood that it was time to search for a new spacious office.

A new office building was found quickly. It was located near a subway station and a nice park. In addition, there was a large underground parking. The number of rooms in the office was the same as the number of projects in the company, so Shiftman decided to assign a room to each project, thus creating a unique work atmosphere for the teams. Project managers had their own notions of ideal room for their projects. Of course, the room should not be too small. However, if the room would be too big, the programmers might be afraid that a new team would be added to their room. Help the managers to assign the rooms quickly and without the boss’s meddling.

### Input

The first line contains the number  $n$  of projects in the company ( $2 \leq n \leq 1000$ ). In the second line you are given  $n$  numbers, which are the areas of all the rooms in the new office. The  $i$ th of the following  $n$  lines contains two numbers, which are the minimum and maximum areas of the room in which the team of the  $i$ th project agrees to work (of course, the minimum area does not exceed the maximum area). All the areas are positive integers and do not exceed 10000.

### Output

If there is only one way to assign the rooms to the teams, output “Perfect!” in the first line and a permutation of integers from 1 to  $n$  in the second line. In this permutation, the  $i$ th element must be the number of the room assigned to the team of the  $i$ th project. The rooms are numbered from 1 to  $n$  in the order in which they are described in the input. If there are several ways to assign the rooms, output “Ask Shiftman for help.” If it is impossible to assign the rooms as required, output “Let’s search for another office.”

### Examples

<i>standard input</i>	<i>standard output</i>
3 40 50 60 30 70 20 40 60 60	Perfect! 2 1 3
3 40 50 70 30 70 20 50 60 60	Let’s search for another office.

## Problem L. Dick and his forwards (Division 2 Only!)

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 64 MiB

Roman and Pavel are playing next game. Roman is told by their coach Dick to walk forward  $a$  steps ( $1 \leq a \leq 100$ ) and then walk backward  $b$  steps ( $1 \leq b \leq 100$ ), after which he repeats  $a$  forward,  $b$  backward, etc. Likewise, Pavel is told to walk forward  $c$  steps ( $1 \leq c \leq 100$ ) and then walk backward  $d$  steps ( $1 \leq d \leq 100$ ), after which he repeats  $c$  forward,  $d$  backward, etc. You may assume that  $a \geq b$  and  $c \geq d$ .

Roman and Pavel have the same length of step, and they are required to take their steps simultaneously (that is, Roman and Pavel will both step forward on their first steps at the same time, and this will continue for each step).

Roman and Pavel start walking from one end of a soccer field. After  $s$  steps ( $1 \leq s \leq 10^4$ ), Dick will blow the whistle. Your task is to figure out who has moved the farthest away from the starting position when the whistle is blown.

### Input

The input will be the 5 integers  $a$ ,  $b$ ,  $c$ ,  $d$ , and  $s$ , each on a separate line.

### Output

The output of your program will be one of three possibilities: “**Roman**” if Roman is farther ahead after  $s$  steps are taken; “**Pavel**” if Pavel is farther ahead after  $s$  steps are taken; “**Tied**” if Roman and Pavel are at the same distance from their starting position after  $s$  steps are taken.

### Example

<i>standard input</i>	<i>standard output</i>
4 2 5 3 12	Pavel

## Problem M. Kanga and Roo (Division 2 Only!)

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 64 MiB

Baby Roo is learning to count on his fingers. When Kanga tells her a number  $n$  ( $1 \leq n \leq 10$ ), he asks “What is  $n$ , Kanga?”, by which he means “How many fingers should I hold up on each hand so that the total is  $n$ ?”

To make matters simple, Kanga gives him the correct finger representation according to the following rules:

- the number may be represented on one or two hands;
- if the number is represented on two hands, the larger number is given first.

For example, if Roo asks “What is 4, Kanga?”, Kanga may reply:

- 4 is 4.
- 4 is 3 and 1.
- 4 is 2 and 2.

Your job is to make sure that Kanga gives the correct number of answers.

### Input

The input will be a single integer  $n$  in the range between 1 and 10.

### Output

The output is the number of ways of producing that number on two hands, subject to the rules outlined above.

### Example

<i>standard input</i>	<i>standard output</i>
4	3