An easy geometry

Time Limit : 3000/1000ms (Java/Other) **Memory Limit : 65535/32768K (Java/Other)**
Total Submission(s) : 0 **Accepted Submission(s) : 0**

Problem Description

There is tetrahedron, four vertex are A,B,C,and D, and you know the length of its six edges,output the volume.

Input

There are several cases.
In each case, there are six positive integers meaning the length of the edges，AB, AC, AD, BC, BD, CD.

Output

Output the volume in two decimal. Output "-1.00" if can't form a tetrahedron.

Sample Input

```
1 1 1 1 1 2
5 6 7 8 9 10
```

Sample Output

```
-1.00
34.18
```

Author

WBN

Ball

Time Limit : 3000/1000ms (Java/Other) **Memory Limit : 65535/32768K (Java/Other)**
Total Submission(s) : 0 **Accepted Submission(s) : 0**

Problem Description

Xiao Ming and his friends like play ball games --Throw the ball.
First of all,they are standing in a row,and then throw the ball between each other.
They appointed:A can throw the ball to B if and only if the people between A and B is shorter than A. But some people are not the really player , they don't play it.Xiao Ming decides to find out these player,so he asks everyone that they throw the ball to which people,and all the friend also answer one by one,if he don’ t play,he would be lie.Now ask you to determine how many player lied.

Input

There are multi case.in each case ,first line contains one integer N(1<=N<=1000), second line contains N integer H[1],H[2],,,,H[N].(1<=H[i]<=500000),H[i] indicates the height of ith person,and H[i]!=H[j](1<=i,j<=N,i!=j).The third line contains N integer V[1],V[2],,,V[N](1<=V[i]<=N) ,V[i] indicates the ith person throw a ball to the V[i]th person.

Output

In each case output an integer how many people lied in a line.

Sample Input

```
5
1 3 4 5 2
3 1 4 1 2
```

Sample Output

```
2
```

Author

zlbing

Binary_tree

Time Limit : 3000/1000ms (Java/Other) **Memory Limit : 65535/32768K (Java/Other)**

Total Submission(s) : 0 **Accepted Submission(s) : 0**

Problem Description

We can use the following method to numbers the binary trees as follows:
(1) empty tree numbered 0
(2) Only one tree node number is 1
(3) For any non-negative integers m, m-node contains the tree’s number is smaller than (m + 1) nodes tree’s ;
(4) comprises a binary tree of m nodes, assuming that the left subtree’s number is L, the right subtree’s is R, and the number of the tree is n, if and only if, all numbers greater than n and contains m nodes binary tree satisfies the following conditions are as follows: its left subtree number greater than L or whose number is equal to the left subtree L, and right subtree Numbers > R

Input

  Multiple test cases , for each test case , given a number n (1 <= n <= 1000000000)

Output

Binary Tree's left child and right child recursive representation. That single node with “x” said that if the binary only left child L, will have to be expressed as (L) x binary only if the right child R, will have to be expressed as X (R) Otherwise, the left sub-tree must be said, that means a (L) X (R).
for each test case, print “Case #X:” in a line , and print the tree in the next line.

Sample Input

```
1
6
20
```

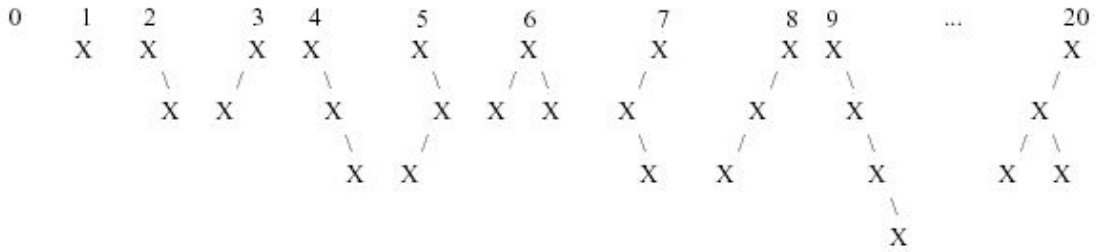
Sample Output

```
Case #1:
x
Case #2:
(x)x(x)
```

Case #3:
((x)x(x))x

Author

goagain



Dice_roll

Time Limit : 3000/1000ms (Java/Other) **Memory Limit : 65535/32768K (Java/Other)**
Total Submission(s) : 0 **Accepted Submission(s) : 0**

Problem Description

There is a dice which have N sides, of course 3-sides polyhedron is not exist, so, you can assume N is larger than 3. Now ,I want to roll the dice, I will stop rolling the dice when and only when I get a number larger than or equal previous number. Thus,

Input

Multiple test cases , for each test case , given a number n (4 <= n <= 10000)

Output

for each test case, print “Case #X:” firstly , and print the answer . (keep 6 numbers after decimal point)

Sample Input

```
4
6
```

Sample Output

```
Case #1:
2.441406
Case #2:
2.521626
```

Author

goagain

Divide groups

Time Limit : 3000/1000ms (Java/Other) **Memory Limit : 65535/32768K (Java/Other)**
Total Submission(s) : 0 **Accepted Submission(s) : 0**

Problem Description

Now, 2*N students numbered from 0 to 2*N-1, want to play a game. In this game, they are divided into two groups A and B,
two students who are corresponding to each other can\'t join the same group. 0 and 1, 2 and 3, 4 and 5,..... 2*N-2 and 2*N-1 are corresponding to each other. And there are also two kinds of conditions.

1 c d (c is the ID of a student, d can be A or B) student whose ID is c can only join group d
2 c e (c and e are differernt ID of two students) students whose ID are c and e must be in the same group

Can you tell whether the game can go on?

Input

There are several cases, the first line is N (N<=10000) and M（M<=10000), there are 2*N students playing the game and
M conditions as mentioned above.

Output

Output \"OK\" if the game can go on.
Output \"NO\" if the game can\'t going on.

Sample Input

```
1 2
1 0 A
1 0 B
2 2
1 1 B
2 3 1
```

Sample Output

```
NO
OK
```

Author

WBN

Gun

Time Limit : 3000/1000ms (Java/Other) **Memory Limit : 65535/32768K (Java/Other)**
Total Submission(s) : 0 **Accepted Submission(s) : 0**

Problem Description

Xiao Ming have a gun which can absorb energy at the speed of one unit in a minute.When the gun's energy is greater than or equal to C,he can shoot the enermy and cost the energy of C. At the same time, the gun's damage is changing along with the time.

Input

There are multiple case.
First line contains two integers T and C(0<=T,C<=100000).T is time and C as described above.
Second line contains T integers a[1],a[2]…a[T](a[i] indicates gun’s damage at ith time)

Output

Output the maximum sum of the damage of the gun shooting for whole T times

Sample Input

```
3 3
1 2 3
```

Sample Output

```
3
```

Author

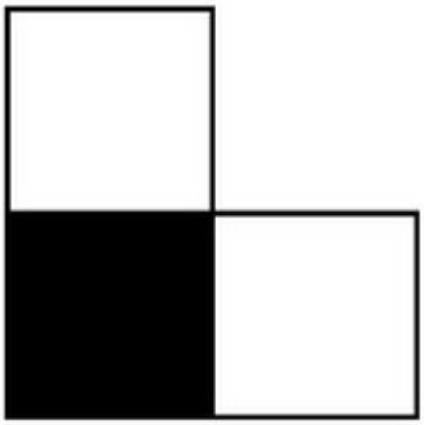
zlbing

L-shape

Time Limit : 10000/5000ms (Java/Other) **Memory Limit : 65535/32768K (Java/Other)**
Total Submission(s) : 0 **Accepted Submission(s) : 0**

Problem Description

As we all know,many people like playing Tetris.Now,we specify a symbol "L" which consists of three squares.Among them,the top one and the right one are white coating,the one which in the lower left is black . we give you a graph n*m and ask if the Tetris could be made up by such "L" or not. The number of "L" is infinit and can be rotated.



Input

On the first line a positive integer T(1<=T<=100) which is the number of test cases
After that per test case:
One line contain two positive integer n,m indicating n rows and m cols
Next n lines ,each line contain m symbols.(n,m<=500) ‘B’ is black,’ W’ is white，’ .’ Is nothing.

Output

For each test case output answer ‘YES’ or ‘NO’ in a line.

Sample Input

```
3
3 4
BWW.
```

```
WWBW
..WB
3 3
W..
BW.
WBW
4 4
.WW.
WBBW
WBBW
.WW.
```

Sample Output

```
YES
NO
YES
```

Author

zlbing

Math Math Math I

Time Limit : 3000/1000ms (Java/Other) **Memory Limit : 65535/32768K (Java/Other)**
Total Submission(s) : 0 **Accepted Submission(s) : 0**

Problem Description

Math is important for ACMer.So this si a math problem.
We know that a prime number is a number that is is only evenly divisible by 1 and itself. The first prime numbers are 2,3,5,7 but they quickly become less frequent. One of the hard questions is how dense they are in various ranges. Adjacent primes are two numbers that are both primes, but there are no other prime numbers between the adjacent primes. For example, 2,3 are the only adjacent primes that are also adjacent numbers.
Your program is given 2 numbers: L and U (1<=L< U<=2,147,483,647), and you are to find the two adjacent primes C1 and C2 (L<=C1< C2<=U) that are closest (i.e. C2-C1 is the minimum). If there are other pairs that are the same distance apart, use the first pair. You are also to find the two adjacent primes D1 and D2 (L<=D1< D2<=U) where D1 and D2 are as distant from each other as possible (again choosing the first pair if there is a tie).

Input

Each line of input will contain two positive integers, L and U, with L < U. The difference between L and U will not exceed 1,000,000.

Output

For each L and U, the output will either be the statement that there are no adjacent primes (because there are less than two primes between the two given numbers) or a line giving the two pairs of adjacent primes.

Sample Input

```
2 17
14 17
```

Sample Output

```
2,3 are closest, 7,11 are most distant.
There are no adjacent primes.
```

Author

LJP

Math Math Math II

Time Limit : 3000/1000ms (Java/Other) **Memory Limit : 65535/32768K (Java/Other)**
Total Submission(s) : 0 **Accepted Submission(s) : 0**

Problem Description

Math is important for ACMer.So this is another math problem.
We konw that the factorial function, n! is defined thus for n a non-negative integer:
0! = 1

n! = n * (n-1)! (n > 0)

We say that a divides b if there exists an integer k such that
k*a = b

Input

The input to your program consists of several lines, each containing two non-negative integers, n and m, both less than 2^31.

Output

For each input line, output a line stating whether or not m divides n!, in the format shown below.

Sample Input

```
6 9
6 27
20 10000
20 100000
1000 1009
```

Sample Output

```
9 divides 6!
27 does not divide 6!
10000 divides 20!
100000 does not divide 20!
1009 does not divide 1000!
```

Author

LJP

Barn Repair

Problem Description

It was a dark and stormy night that ripped the roof and gates off the stalls that hold Farmer John's cows. Happily, many of the cows were on vacation, so the barn was not completely full.

The cows spend the night in stalls that are arranged adjacent to each other in a long line. Some stalls have cows in them; some do not. All stalls are the same width.

Farmer John must quickly erect new boards in front of the stalls, since the doors were lost. His new lumber supplier will supply him boards of any length he wishes, but the supplier can only deliver a small number of total boards. Farmer John wishes to minimize the total length of the boards he must purchase.

Given M (1 <= M <= 50), the maximum number of boards that can be purchased; S (1 <= S <= 200), the total number of stalls; C (1 <= C <= S) the number of cows in the stalls, and the C occupied stall numbers (1 <= stall_number <= S), calculate the minimum number of stalls that must be blocked in order to block all the stalls that have cows in them.

Print your answer as the total number of stalls blocked.

Input

Line 1: M, S, and C (space separated)
Lines 2-C+1: Each line contains one integer, the number of an occupied stall.

Output

A single line with one integer that represents the total number of stalls blocked.

Sample Input

```
4 50 18
3
4
6
```

```
8
14
15
16
17
21
25
26
27
30
31
40
41
42
43
```

Sample Output

```
25
[One minimum arrangement is one board covering stalls 3-8, one covering 14-21, one covering 25-31, and one covering 40-43.]
```

Author

goagain