Problem 1

Validate an Input Password

Given a password from the keyboard, validate it, taking into account the following rules:

1. Password must have 5 to 17 characters.
2. At least one numeric digit and no more than 4 digits.
3. At least one capital letter.
4. Starting character must not be a number.
5. Must not have space or slash (/)
6. At least one alphabet
7. 2 similar digits must not appear consecutively

Sample Input	Sample Output
Satanas567	Valid password
4ElTemible	Invalid password
manossangrientas9	Invalid password
Satanas666	Invalid password

Problem 2

Parallelogram Text

1. Given a text from the keyboard, represents a parallelogram with it.

Sample Input	Sample Output
GATO	G GA GAT GATO ATO TO O
CONEJO	C CO CON CONE CONEJ CONEJO ONEJO NEJO EJO JO O

Problem 3

Christmas Trees

Children do love draw Christmas trees and you challenged some of them to draw trees of many sizes just with the asterisk "*" char.

The rule is simple: from bottom to top, the trunk of the tree consists of one line with three asterisks and one line with one asterisk. Then comes the rest of the tree, with each row starting with the size ordered by you and decreasing every two character, until you reach the top of the tree that have just one asterisk. Note that for this to work, will only be allowed odd sizes to the trees.

Input
The input consists of an integer number N (2 < N < 100).
Output
For that number N, print a tree as specified above and as the example below, with a blank line after each tree.

For example:

Input Sample	Output Sample
| 9 | ```
        *
      * * *
    * * * * *
  * * * * * * *
* * * * * * * * *
        *
      * * *
``` |
| 5 | ```
    *
  * * *
* * * * *
    *
  * * *
``` |

Problem 4

The Gary's Series

Gary is in charge of developing an animation for biologists, as Gary wants to surprise them, he has thought of building an algorithm where the petals of a flower are formed, he wants to imitate nature and knows that he can achieve it by taking advantage of the horizontal space, he has heard about a sequence called "the divine proportion" which is what can help him to achieve his goal.

This sequence is present in absolutely everything, for example, sunflowers contain 21 spirals that are directed in one direction and 34 that go in the opposite way, both figures are consecutive in this chain. Another example is present in pine pineapples, which have two groups of spirals that go in opposite directions and the proportion they keep between the number of one and the other is represented by sequential numbers of the series 0, 1, 1, 2, 3,…, N

Gary knows that the sequence he is looking for involves the numbers from the previous series, so the series he needs is the following: 1/1, ½, 2/3 ...... 144/233

Gary asks you to build the algorithm, that given a number indicate if it belongs to the sequence and which is the successor.

Input
The input consists of several test cases. Each case test is a number N, where $0<N<1346269$

Output:
For each test case, print if the number belongs to the series and which is the successor. Print "The number is not valid", otherwise

Input Sample	Output Sample
2	The number is valid in the sequence of the series and is 2 / 3. The successor is 3 / 5
4	The number is not valid
832040	The number is valid in the sequence of the series and is 514229 / 832040. The successor is 832040 / 1346269

Problem 5

Longest Well-Formed Parentheses String

Given a string containing characters and symbols. You must detect in that given string, only the characters '(' and ')', and you must find the length of the longest valid (well-formed) parenthesized substring. Note: the complete "Longest Well-Formed Parentheses String count" will include all internal symbols between the parentheses.

For example:

Input Sample	Output Sample
Efecto(sonriendo(computo)	9
50Perros)23 gatos(6+50)-(fantasma54))	19

Explanation output sample 1: The Longest Well-Formed Parentheses String is: (computo)

Explanation output sample 2: The Longest Well-Formed Parentheses String is: (6+50)-(fantasma54)

Problem 6

Fewest Coin Combinations

You have a set of coins of different denominations and a total amount of money amount. Write a program to compute the fewest number of coins that you need to make up that amount. If that amount of money cannot be made up by any combination of the coins, return -1.

Note:
You may assume that you have an infinite number of each kind of coin.

For example:

Input Sample	Output Sample
Coins = [1,2,5], amount = 11	3
Coins = [2], amount = 3	-1

Explanation for input 1: 11 = 5 + 5 + 1

Explanation for input 2: it is not possible because you need a coin of 1 to complete 3