Marks	**Set D**
15 marks	**Combination Lock**
10 marks	**Pascal's Triangle**
10 marks	**Cryptic Lab Notes**
TOTAL: 35 marks	

Question1: Combination Lock

There is a combination ***lock*** and a ***phrase*** engraved on the door to a highly access restricted area. The ***lock*** can be opened using a specific combination of letters of alphabet. The ***phrase*** contains between five to twelve distinct letters of alphabet. Five of the letters from the given ***phrase*** form the combination that opens the safe. A positive integer, called the ***key***, is needed along with the engraved ***phrase*** to find out the combination. Each person who is entitled to enter the door is given a new ***key*** for each entry. Applying a rule on the engraved ***phrase*** and the ***key***, the person can find the combination to the lock.

The rule to find the combination

Select any five letters v, w, x, y, and z from the engraved phrase on the door that satisfy the following equation: $v - w^2 + x^3 - y^4 + z^5 =$ ***key***, where each letter is replaced by its ordinal position in the alphabet (A=1, B=2, ..., Z=26). The combination is then vwxyz. If there are more than one solutions then the combination is the one that is lexicographically greatest, i.e., the one that would appear last in a dictionary.

Your job is to write a program which will be able to find the door combination given the ***key*** and the ***phrase*** as input. There is a possibility that for some inputs there may be no combination. In such a case, you need to print 'no solution'. Assume all letters of alphabet are only upper case.

An example:

Given ***key*** as 1 and ***phrase*** as ABCDEFGHIJKL, one possible combination is FIECB, since $6 - 9^2 + 5^3 - 3^4 + 2^5 = 1$ (F=6, I=9, E=5, C=3, B=2). There are actually several solutions in this case, and the lexicographically greatest one turns out to be LKEBA.

Your program will have the following inputs and output:

Input:
key phrase

[0<key<12,000,000
phrase: at least five and at most twelve distinct uppercase letters]

Output:
Combination or 'no solution'

Example Input	Example Output
1 ABCDEFGHIJKL	LKEBA
11700519 ZAYEXIWOVU	YOXUZ
3072997 SOUGHT	GHOST
1234567 THEQUICKFROG	no solution

Note: You answer has to be embedded within '[' and ']'.

Question2: Pascal's Triangle

A Pascal's triangle is a number pattern that is often represented as a triangular pattern shown in Figure 1. It starts with '1' at the top or the zeroth row. Every number in the triangle is the sum of the two numbers diagonally above it to the left and the right. We have highlighted one such number in the Figure 1. The left and the right most edges are '1'.

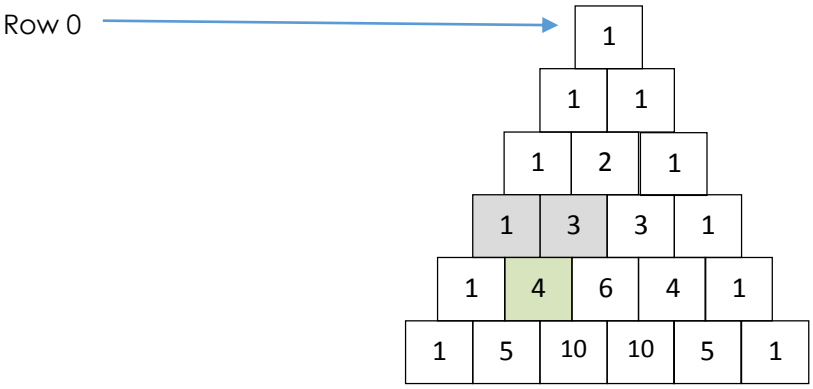

Figure 1. Representation of Pascal's triangle

Assume that in the Pascal's triangle you have inversed all the numbers in each row (1 divided by the number). For example, numbers in the 3rd row will be 1/1, 1/3, 1/3, 1/1 instead of 1, 3, 3, 1, and so on. Further, as shown in Figure 2, the Pascal's triangle under consideration has its two outermost edges of numbers removed.

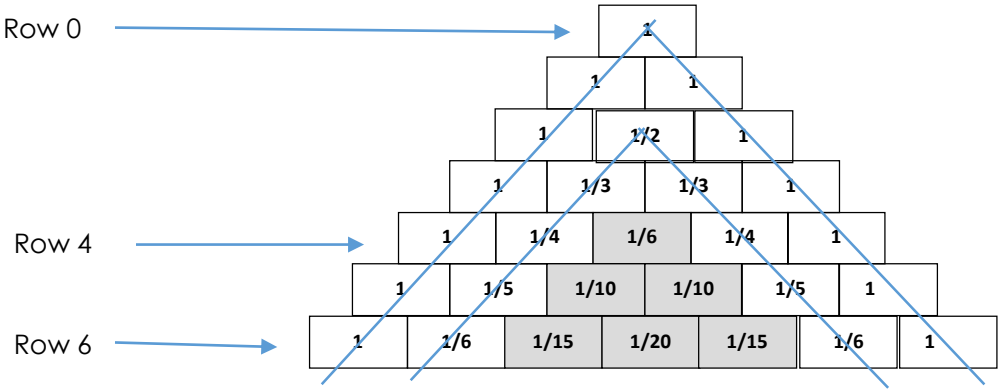

Figure 2. Inversed numbers in a Pascal's triangle with 2 outer edges removed

Write a program that for a given input row r (read from the command line), prints to the screen the summation s (rounded to one digit after decimal point) of the inversed numbers starting from row 4 to row r for a triangle. Assume r <= 1000.

An example:

Say, r=5, then s=1/6 + (1/10 + 1/10) = 0.366 → [0.4], where row 4→ 1/6 and row 5 -> (1/10,1/10).

Your program will have the following inputs and outputs:

Input:
r

where 4<=r<=1000

Output: s
[Note: s should be rounded to one digit after decimal point]

Example Input	Example Output
3	[0.0]
4	[0.2]
5	[0.4]
6	[0.6]

Note: You answer has to be embedded within '[' and ']'. Outputs should be rounded to one digit after the decimal point.

Question3: Cryptic Lab Notes

You have joined a top secret defense project to continue the work of a notable scientist who left the work mid-way. You have been tasked with interpreting the encrypted experiment results kept by the scientist. The experiment details and encryption rule is described below:

Each set of experiments has ***n*** phases. Value of measurement taken in each phase is a positive integer <= ***k***. The measurements are monotonically increasing, that is, each measurement would be at least as big as all that precede it. For example, here is a set of measurements for one such experiment with ***n***=*13* and ***k***=6:

1, 1, 2, 2, 2, 2, 2, 4, 5, 5, 5, 5, 6

Note that ***n*** has to be larger than ***k***. Hence, there will be many repeated values in one set of measurement.

The encryption: The scientist recorded the above experiment as set ***P*** with ***k*** values defined as follows: ***P(j)***, where1 ≤ ***j*** ≤ ***k***, denoted the number of phases having a measurement of j or less. For example, the original measurements from the above experiment were recorded as the ***P***-sequence:

2, 7, 7, 8, 12, 13

Explanation: Two measurements <=1 (P(1)=2), seven measurements <=2 (P(2)=7), seven measurements <=3 (P(3)=7), eight measurements <=4 (P(4)=8), twelve measurements <=5 (P(5)=12), thirteen measurements <=6 (P(6)=13). Your job is to write a program that recovers the original measurements for the experiments from these P-sequences.

Your program will have the following inputs and output:

Input:
k P-sequence

where k is the length of the P-sequence. Following that are k values of the P-sequence.
[1 ≤ k < n ≤ 26.]

Output:
One line containing the original experiment measurements separated by spaces.

Example Input	Example Output
6 2 7 7 8 12 13	[1 1 2 2 2 2 2 4 5 5 5 5 6]
1 4	[1 1 1 1]
3 4 4 5	[1 1 1 1 3]

Note: You answer has to be embedded within '[' and ']'. Outputs should be whole numbers and not be represented as decimals.