ACM 2017: Star Arrangements

The recent vote in Puerto Rico favoring United States statehood has made flag makers very excited. An updated flag with 51 stars rather than the current one with 50 would cause a huge jump in U.S. flag sales. The current pattern for 50 stars is five rows of 6 stars, interlaced with four offset rows of 5 stars. The rows alternate until all stars are represented.

```
* * * * * *
 * * * * *
* * * * * *
 * * * * *
* * * * * *
 * * * * *
* * * * * *
 * * * * *
* * * * * *
```

This pattern has the property that adjacent rows differ by no more than one star. We represent this star arrangement compactly by the number of stars in the first two rows: `6,5`.

A 51-star flag that has the same property can have three rows of 9 stars, interlaced with three rows of 8 stars (with a compact representation of `9,8`). Conversely, if a state were to leave the union, one appealing representation would be seven rows of seven stars (`7,7`).

A flag pattern is visually appealing if it satisfies the following conditions:

- Every other row has the same number of stars.
- Adjacent rows differ by no more than one star.
- The first row cannot have fewer stars than the second row.

Your team sees beyond the short-term change to 51 for the US flag. You want to corner the market on flags for any union of three or more states. Given the number *S* of stars to draw on a flag, find all possible visually appealing flag patterns.

Input Format

The input consists of a single line containing the integer *S*.

Constraints

3 <= *S* <= 32 767

Output Format

On the first line, print *S*, followed by a colon. Then, for each visually appealing flag of *S* stars, print its compact representation, one per line.

This list of compact representations should be printed in increasing order of the number of stars in the first row; if there are ties, print them in order of the number of stars in the second row. The cases 1-by-*S* and *S*-by-1 are trivial, so do not print those arrangements.

The compact representations must be printed in the form `x,y`, with exactly one comma between `x` and `y` and no other characters.

Sample Input 0

```
50
```

Sample Output 0

```
50:
2,1
2,2
3,2
5,4
5,5
6,5
10,10
13,12
17,16
25,25
```

Sample Input 1

```
51
```

Sample Output 1

```
51:
2,1
3,3
9,8
17,17
26,25
```

Sample Input 2

```
32767
```

Sample Output 2

```
32767:
4,3
7,6
7,7
16,15
31,31
36,35
76,75
109,108
151,151
217,217
462,461
529,528
1057,1057
2341,2340
2521,2520
4681,4681
16384,16383
```