7269 Snake Carpet

In school of EECS of Peking University, there is a homework for all freshman — the contest of AI snakes. This contest is ended today. Bacchus has got a very good result, so he decides to make a carpet full of snakes as a souvenir, and lays it over the floor in his room.

As his room is square, a square carpet is needed. A $H \times W$ carpets is made up of $H \times W$ *units (each unit is* 1×1*). Snakes can have different length, but all snakes' width is 1 unit.* For some reason, he hopes that N special snakes are drawn on the carpet: the length of the i-th snake should be i, which can be seen as i connected units (Two units that share an edge are considered connected). Except the first snake, the $(2k-1)$-th snake should have positive odd number of turning points; except the second snake, the $2k$-th snake should have an positive even number of turning points. i and k both start from 1. Each snake should not intersect with itself, nor with other snakes. All units of the carpet must be covered by snakes.

But the question is whether there is a solution.

Input

Multiple test cases. There will be up to 25 cases.

For each test case: one line contains one integer N, indicating the number of snakes. $(1 \leq N \leq 500)$

Output

For each test case:

If the solution does not exist, output one line '`0 0`', otherwise output $N + 1$ lines: The first line contains two integers H and W, indicating the height and the width of the carpet. You should guarantee that $H \times W = 1 + 2 + \ldots + N$. For the next N lines, the i-th line contain $2i$ integers, indicating the coordinates of the i-th snake in order. The coordinate of top-left corner unit is $(1, 1)$ and the coordinate of bottom-right corner unit is (H, W).

Hint: This problem is special judged, and the solutions for the sample input are on the right:

3 1 2
3 3 2

3 4 4 1 2
3 3 4 4 2

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

Sample Input

```
3
4
5
```

Sample Output

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

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