Lovely Triplets

by nikasvanidze

Problem | Submissions | Leaderboard | Discussions | Editorial

Daniel loves graphs. He thinks a graph is *special* if it has the following properties:

- It is undirected.
- The length of each edge is 1.
- It includes *exactly* P different *lovely triplets*.

A *triplet* is a set of 3 different nodes. A triplet is *lovely* if the minimum distance between each pair of nodes in the triplet is *exactly* Q. Two triplets are different if 1 or more of their component nodes are different.

Given P and Q, help Daniel draw a *special graph*.

Input Format

A single line containing 2 space-separated integers, P (the number of different lovely triplets you must have in your graph) and Q (the required *distance* between each pair of nodes in a lovely triplet), respectively.

Constraints

- $1 \leq P \leq 5000$
- $2 \leq Q \leq 9$

Output Format

For the first line, print 2 space-separated integers, N (the number of nodes in the graph) and M (the number of edges in the graph), respectively. On each line i of the M subsequent lines, print two space-separated integers, u_i and v_i, describing an edge between nodes u_i and v_i.

Your output must satisfy the following conditions:

- $0 \leq N, M \leq 100$
- $1 \leq u_i, v_i \leq N$

If there is more than one correct answer, print any one of them.

Sample Input

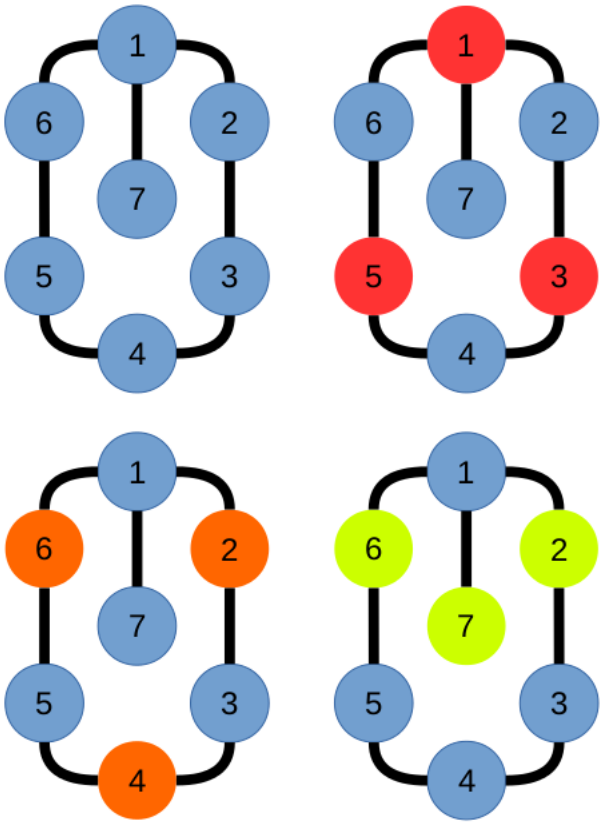
```
3 2
```

Sample Output

```
7 7
1 2
2 3
3 4
4 5
5 6
6 1
1 7
```

Explanation

There are exactly $P = 3$ lovely triplets in this graph: $\{1, 3, 5\}$, $\{2, 4, 6\}$, and $\{2, 6, 7\}$.



Observe that each node in a lovely triplet is $Q = 2$ edges away from the other nodes composing the lovely triplet.

Solved score: 80.00pts

Submissions: 340

Max Score: 80

Difficulty: Advanced

Rate This Challenge:

More

Current Buffer (saved locally, editable) Java 7

```
1  import java.io.*;
2  import java.util.*;
3  import java.text.*;
4  import java.math.*;
5  import java.util.regex.*;
6
7  public class Solution {
8
9      public static void main(String[] args) {
10         Scanner in = new Scanner(System.in);
11         int P = in.nextInt();
12         int Q = in.nextInt();
13     }
14 }
15
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature