Niccoriˆˆ Survey Team

Problem

In their odyssey to investigate all the smiles in the world, Rin and Ren have arrived to Ecatepec. Ecatepec can be seen as a graph with N vertices, numbered from 0 to $N-1$, connected by some bidirectional edges. However, since it is largely uncharted territory, the team does not know about its edges. If Ecatepec turns out to be not connected[1], Rin and Ren will not be able to carry out their mission properly, so they enlist the help of the great scientist Miku. Miku gives our adventurers a curious device that answers questions as follows given a parameter K that Miku has set beforehand:
Given two distinct vertices of the Ecatepec graph, the device answers whether they are at a distance[2] less than K, exactly K, or greater than K.
Since the device is still a prototype, Miku warns the team that they can ask at most $\frac{2N^2}{K}$ questions. Rin and Ren failed math in high school, so they enlist your help to determine if Ecatepec is connected.

Implementation Details

You must implement the function *Equipo_sonrisas()*. This function receives an integer N (the size of the graph) and an integer K (the device's parameter). This function should return a pair of integers: if the graph is connected, it should return the pair $(-1, -1)$. If the graph is not connected, it should return a pair of vertices in different components.
During your program, you can call the function *Dispositivo_Miku()*. This function receives 2 integers a, b and returns -1 if $dist(a, b) < K$, 0 if $dist(a, b) = K$, and 1 if $dist(a, b) > K$. To call this function, you must include the library *"Sonrisas.h"* with the command *#include "Sonrisas.h"*. An example program would look like this:

```
#include "Sonrisas.h"
#include <bits/stdc++.h>
using namespace std;

pair<int, int> Equipo_sonrisas(int N, int K) {
    // Implement this function.
}
```

The grader will call this function **multiple** times for each testcase.

[1] We say that a graph is connected if for every pair of vertices, there is a path (a sequence of vertices such that each pair of adjacent vertices is connected by an edge) that starts at one and ends at the other.

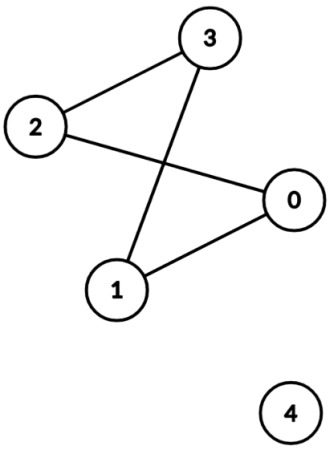
[2] The distance between two vertices is defined as the path with the fewest edges connecting them, expressed as $dist(a, b)$.

Example

- The grader calls the function

  Equipo_sonrisas(5, 2)

  The hidden graph is as follows:



- the following is a distance table of the graph:

$dist(a, b)$	0	1	2	3	4
0	0	1	1	2	∞
1	1	0	2	1	∞
2	1	2	0	1	∞
3	2	1	1	0	∞
4	∞	∞	∞	∞	0

- Here is an example interaction:

Function called	return value
Dispositivo_Miku(0, 1)	-1
Dispositivo_Miku(0, 2)	-1
Dispositivo_Miku(0, 3)	0
Dispositivo_Miku(0, 4)	1
Dispositivo_Miku(1, 2)	0
Dispositivo_Miku(1, 3)	-1
Dispositivo_Miku(1, 4)	1
Dispositivo_Miku(2, 3)	-1
Dispositivo_Miku(2, 4)	1
Dispositivo_Miku(3, 4)	1

- If the return value of the function called by the grader is $(-1, -1)$, the veredict would be wrong answer. Otherwise, if the return value is, for example, $(1, 4)$, the veredict would be accepted.

Constraints

- $1 \leq K < N \leq 1000$.
- In all cases, the number of times you call the function *Dispositivo_Miku()* must be less than $\frac{2N^2}{K}$. Otherwise, you will receive a wrong answer veredict.
- If the function returns $(-1, -1)$ and the graph is not connected, or if it returns a pair of connected nodes, you will receive a wrong answer veredict.
- Let S_N be the sum of the values of N for each call made by the grader in a testcase. It is guaranteed that $S_N \leq 1000$.

Subtasks

- (5 points) $K = N - 1$.
- (10 points) $K \leq 4$.
- (33 points) $K = \frac{N}{2}$.
- (22 points) The graph is a forest (there are no cycles, that is, there is no paths that don't repeat edges and start and end at the same vertex).
- (30 points) No additional constraints.