Programming assignment for home (after submitting the first 6 tasks)

You have just discovered a new underground pyramid, clearly the result of a cooperation between ancient Egyptians and a super developed alien race. In the largest room, when you look up, you see some very peculiar objects fixed to the ceiling. There are black metal cubes with one face of each pressed against the ceiling. They are connected by shiny solid platinum rods (some straight, some curved). The connections of the rods and the cubes are fully rigid - they are welded together or something. The rods are clearly never fixed to the ceiling. But, at least some of the cubes must be, as this is what keeps the whole thing from falling down. After some probing, you figure out that the cubes are miniature anti-matter bombs with motion sensors. If any of those were to fall, it would be the end of the solar system. Still, you are willing to take your chances: we are talking about kilograms and kilograms of platinum in those rods!! You want to cut down as many of the rods as possible without risking any of the cubes to fall. The good news is that those rods are so strong that potentially one is able to bear the weight of even all the cubes. The bad news is that there is no way to figure out which cubes are fixed to the ceiling.

Task

- **Write a Python program** which can tell you how many rods you can cut down. (You only need to provide the number, no need to enumerate which ones to cut.) Make sure your solution works for all corner cases. The input is the layout of the rods and cubes. Cubes are identified by integers and each item of the input corresponds to a rod: it tells which two cubes are connected by that rod.
- **Explain the algorithm** you used. How does it work? Why is it correct?

Requirements

- The submission has to be written in Python3.
- The solution should implement a function compute(rods) that takes the rods as input and returns the number of rods that can be removed.
  - The argument rods should be a list of 2-tuples, each tuple corresponding to a rod with the cube identifiers being the elements of the tuple. For example the input rods = [(1, 2), (1, 3)] means we have three cubes, identified by 1, 2 and 3, and two rods, one connecting the cubes 1 and 2, and the other connects the cubes 1 and 3.
  - The function compute() should return one value, the number of rods which could be cut. For example compute([(1, 2), (1, 3), (3, 2)]) should return the value 1.
- Attach a short text file in which you explain your algorithm. It's a plus if you can provide proof of correctness.

Example

rods = [(42, 35), (20, 35), (20, 35), (42, 10)]
compute(rods) # Returns 1

This means there are four rods, two between cubes 20 and 35, one between cubes 35 and 42 and one between cubes 10 and 42. The correct output for this input is 1, because you can safely remove one of the rods between 20 and 35.