Beaumont Yin
CS 146

Task: You are a product manager and currently leading a team to develop a new product. Unfortunately, the latest version of your product fails the quality check. Since each version is developed based on the previous version, all the versions after a bad version are also bad.

Suppose you have n versions [1, 2, ..., n] and you want to find out the first bad one, which causes all the following ones to be bad.

You are given an API bool isBadVersion(version) which returns whether version is bad. Implement a function to find the first bad version. You should minimize the number of calls to the API.

Approach:

1. Perform binary search on the given integer
2. If the current middle number is a bad version then look at the left half, other wise look at the right half
3. Repeat this process and return the left number of the binary search

Test cases:

```
public boolean isBadVersion(int n){ // created my own method for testing purposes based on the leet code problem
      return n>=4;
  }

Solution solution = new Solution();
      int n = 10;
      System.out.println(solution.firstBadVersion(n));
```