CS2040S: Data Structures and Algorithms

Contest 1 – Catch the Spies!

1 Background

I think CS2040S has been compromised! We have just received information that hidden among the N students of CS2040S are K spies! Your job is to identify all the spies.

You can send a group of ($\geq K$) students on a mission. If all K spies are on the same mission, they will have a secret meeting to plot their evil plots. For any set of students that are sent on a mission together, you will learn if a secret meeting occurred, or not. You learn nothing else.

As discussed in lecture, it just so happens that 122 is the minimum number of missions needed for $N = 1024, K = 17$. (Can you prove that?)

2 Code Briefing

2.1 Interface Descriptions

Within `code.zip`, you should see the `IMissionControl` interface:

```
public interface IMissionControl {

    public boolean sendForMission(int[] mission);

}
```

The interface provides the following method for you to use in your solution:

- `boolean sendForMission(int[] mission)`:
  Sends a group of students on a mission and returns true if there's a secret meeting, false otherwise.
  `mission` should be an array of length N of 0's and 1's (represented as integers), where a 1 at index i means that student i is sent on the mission. As per the description above, the array needs to have $\geq K$ 1's.
  For example, for $N = 5, K = 2$, `mission` $= \{1, 0, 1, 0, 0\}$ means that students 0 and 2 are sent on the mission.

In the tasks below, your job is to implement the `IFindSpies` interface:

```
public interface IFindSpies {

    public int[] findSpies(int N, int k, IMissionControl missionControl);

}
```

The interface requires you to support the following method:

- `int[] findSpies(int N, int k, IMissionControl missionControl)`:
  Returns an array of length N of 0's and 1's (represented as integers), where a 1 at index i means that student i is a spy. As per the description above, the array should have K 1's.
  For example, for $N = 5, K = 2$, $\{1, 0, 1, 0, 0\}$ means that students 0 and 2 are spies.
  You are given `N`: the total number of students, `k`: the total number of spies, and `missionControl` as described above for you to call `sendForMission`.

2.2 Testing

To test your implementation, you can run the `main` function in `RunContest.java`. If your implementation is correct (returns the correct spy bitmap), you should see in your console relevant information about the performance of your algorithm. Otherwise, you should see an Exception being thrown.
Feel free to change `bitmap` to experiment with different N and k.

3 Tasks

Problem 1: For any N students and K spies, what is the minimum number of missions that we need? Implement an algorithm in `FindSpyMinimumSteps.java` that attempts to achieve this minimum number of missions.

Bonus Optional: What is the asymptotic upper bound? Can you prove it?

Problem 2: Suppose it costs \$1 to send each student on a mission. What is the minimum amount that we need to spend? Implement an algorithm in `FindSpyLowestCost.java` that attempts to achieve the minimum cost required to identify all K spies.

4 Grading

For both problems, we will run your algorithm on different N and k and take note of
1) The number of successful cases (return correct spy bitmap) and
2) The total number of missions sent or cost incurred for successful runs.