Algorithms
Graph Modeling Project (125 pts)

A detailed description of the maze problem including a figure is provided separately[1]. You may want to look at that first to get a sense of what this project involves! Your assignment is to (1) model the maze problem as a graph (2) use an appropriate graph algorithm that we've encountered in class to solve the problem (3) code the algorithm and (4) provide the output on an example input. **Friendly reminder: This is an individual assignment. The departmental collaboration policy (the document you signed at the start of the semester) will be enforced.**

1 Deliverables

The deliverable will take the form of a PDF file to be submitted on Canvas. If we are not confident your approach works, we may follow up and ask for a demo. Your report must include the following four items.

1.1 Problem Modeling[50 pts]

- [10] Explain how you modeled the problem as a graph (typically, a few sentences)
- [15] Draw enough of the resulting graph to convince us that you have modeled the graph correctly (you don't have to draw the whole graph).
- [10] Identify the graph algorithm needed to solve the problem (one sentence).
- [15] Convince us that this algorithm will actually solve the problem. This means that it will find a path if one exists (typically, a few sentences, depending on your approach).

1.2 Code Submission [30 pts]

Please include your code in the report. 10 points will be allocated for quality and readability of the code. The code should read in the provided input file and print the output.

1.3 Results [45 pts]

Display the output of your program on the input maze provided below. Points will be awarded for finding a correct path and printing it in the required format.

1.4 Extra Credit [5 pts]

You will receive a bonus of upto 5 points if you can show that you have done something above and beyond what was asked. Possible examples include but are not limited to (1) using an algorithmic library (e.g., Boost in C++) to implement the graph functionality (2) augmenting your algorithm

[1] The problem was taken from "MAD MAZES: Intriguing Mind Twisters for Puzzle Buffs, Game Nuts and Other Smart People," by Robert Abbott, Bob Adams, Inc. Publishers, 1990.

with a visualization (3) an experimental comparison between algorithms etc. *Please explicitly include a section in your report titled "Extra Credit." Don't just assume the TA will notice your enhancements.*

2 Input Format

Your code will first read the input from a file, which is specified as follows:

- Line 1 contains two integers, n and m.
- Line 2 contains $n-1$ characters where the ith character represents the color of the ith vertex. The vertex with index n is the exit and has no color.
- Line 3 contains two integers s_1 and s_2, representing the index of the starting vertices of Captain Rocket and Lieutenant Lucky, respectively.
- Each of the next m lines contains two integers a and b and one character c, in that order, representing a corridor with color c from a to b. Note that $a,\ b \in [1, n]$.

The input below corresponds to the provided example where 1 is A, 2 is B, ..., 27 is AA and 28 is the Goal vertex. The four colors are R (red), G (green), Y (yellow) and B (blue).

```
28 40
B R R Y Y G G B G Y R G B Y B G R R R R R G R B G R G
1 2
1 5 R
1 10 G
2 3 Y
2 7 B
3 12 B
3 13 G
4 1 R
5 9 R
6 10 R
6 2 Y
7 11 Y
7 17 R
8 4 G
9 14 B
10 15 R
11 6 B
12 17 Y
13 23 Y
14 8 R
14 20 G
15 14 G
16 10 G
16 14 R
16 18 B
16 19 Y
```

```
17 16 B
18 19 Y
19 13 G
20 21 Y
20 24 R
21 22 B
22 23 Y
23 27 B
24 25 G
24 28 B
25 21 Y
25 28 R
26 25 B
26 28 R
27 26 Y
```

3 Output Format

The output will describe a path with each line describing one step. Each line contains a character (either R or L), followed by a space, then an integer x, indicating that Rocket or Lucky respectively moved to room x.

One possible path may start off as follows

```
L 7    // Lucky goes to G
R 10   // Rocket goes to J
L 11   // Lucky goes to K
R 15   // Rocket goes to O
etc
```