Project 3 — Balloon Juice

Jared Dyreson
jareddyreson@csu.fullerton.edu
Mason Godfrey
mgodfrey@csu.fullerton.edu
California State University, Fullerton

Contents

1 Group Mates

- Jared Dyreson
- Mason Godfrey

2 Introduction

This project focuses on a mission reconnaissance visualizer, showing a bot retrieve a balloon in a deeply nested balloon field. The mission is to traverse from balloon to balloon, leaving markers of where it has been in the process. Our bot has a very simple set of instructions and rules it must adhere to. Those are as follows:

1. The bot must move forward once and cannot repeat such action after the fact. That path is considered dead.
2. The bot can go backwards and follows the same principle as the rule above.
3. Teleportation is not permitted
4. The bot will finally show it's path taken in reverse order, making it's way back to the initial balloon.

In this project, we have the balloon field initialized to an $V \times V$ adjacency matrix, where V is defined as the number of balloons occupying the field (here we have 40).

2.1 Floyd-Warshall Algorithm

The algorithm will find the shortest path given a matrix with weighted edges, where the connection spanning from $u \to v$ signifies an edge or connection. These edge values are then used to find the lengths of said shortest path. By design, the algorithm does not attempt to reconstruct the path but in our rendition it does. The average running time of this algorithm is $\theta(|V^3|)$.

3 Archive Contents

3.1 documentation

- exported/Balloon-Juice.pdf — PDF rendition of our documentation
- Balloon-Juice.tex — LaTeX version of our documentation

3.2 root

- Display.js — All drawing functionality that is conducted by p5
- driver.html — webpage to display all the drawing
- README.md — Small readme about the project

3.3 runtime

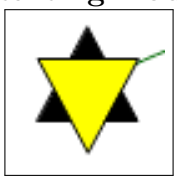
- assets/p5.js — this file is considered an import and has been given by Professor Siska
- backend/Bot.js — contains logic for the bot that traverses the balloon field (remove randomGenerator)
- backend/Grid.js — MARKED FOR REMOVAL
- backend/GenPoint.js — randomly generates balloons in the balloon field, ensuring they have proper DPV values
- backend/Traversals.js — traversal algorithms that are used by the bot
- backend/RNG.js — random number generator class (MARKED FOR REMOVAL)
- graphics/Balloon.js — a balloon and balloon field class implementation that controls the graphical movement of the balloons
- graphics/Cell.js — a graphical representation of the balloons that get displayed onto the screen
- graphics/DPV.js — helps control the logic behind the generation of balloons

4 Installation and Running

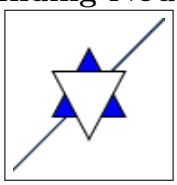
1. Extract the .zip file into a folder
2. Navigate to the driver.html file wih n the folder. Right-click on the file and select "Open". The p5 program should start immediately. A field of balloons should appear and the algorithm should start displaying the path taken.

4.1 Icon Legend

Starting Node



Ending Node



5 Known Bugs

Currently there are no known bugs to exist nor undesired program outcomes.

6 Testing

The team has not experience any warnings or ill side effects from running Balloon-Juice.

7 Credits

Mason:

1. Something

Jared:

1. Floyd-Warshall Algorithm