

Google Maps 0.5



Beeline Pioneers

- Course Instructor: Prof. Clint P. George
- TA : Dr. Chitra Nayagam

By -

- Aniket Chaudhri
- Adarsh Anand
- Rajat Singh

Acknowledgement



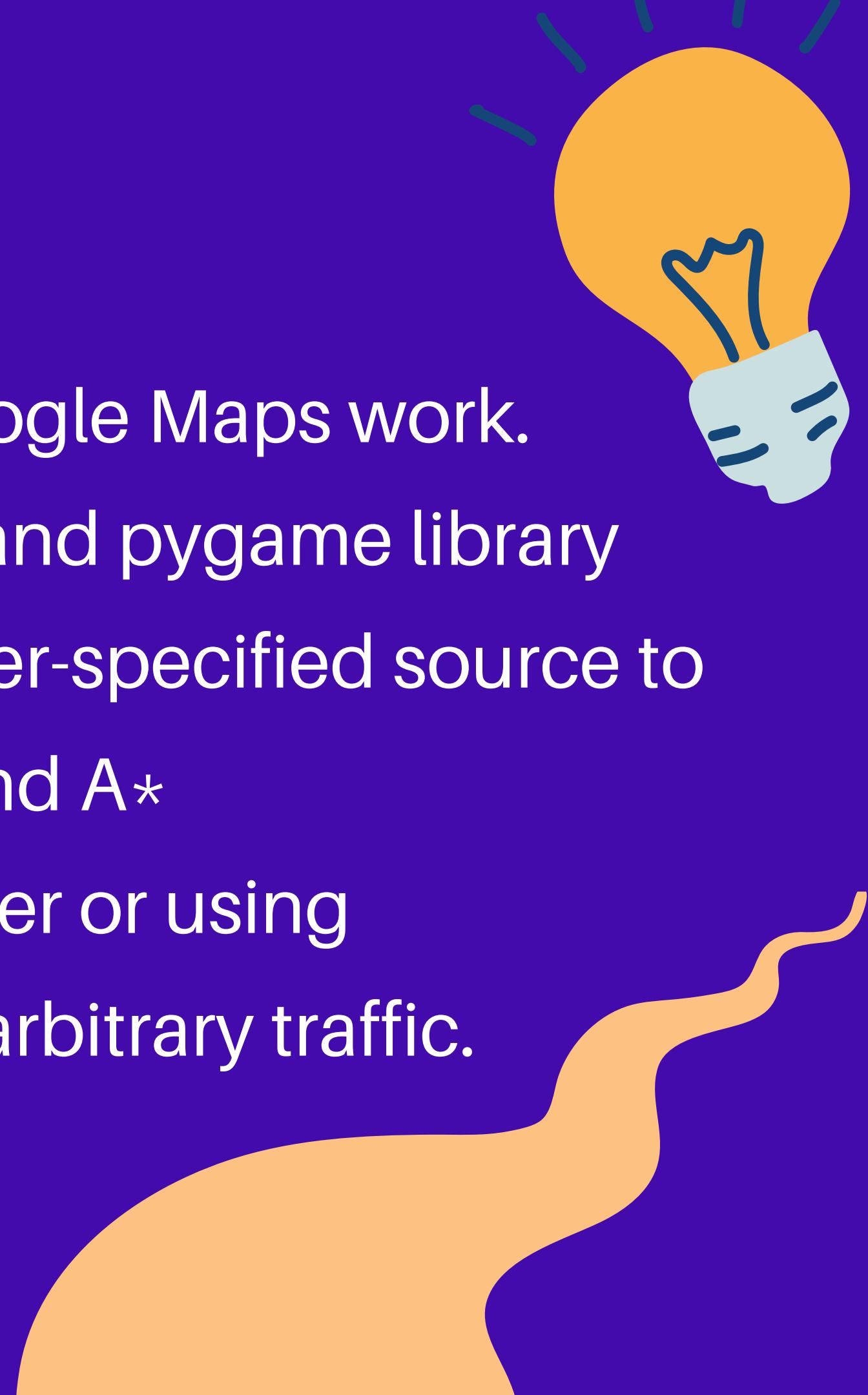
We want to express our gratitude towards Prof. Clint George and Dr Chitra Nayagam for their support in accomplishing our project on Google Maps 0.5

We want to extend deep appreciation to all my group members. Without their support and coordination, we would not have been able to complete this project.

Last but not least, our respect for Google Maps' Developers without whom you wouldn't reach your favourite destinations.

IDEA

- In this project, we aim to showcase how Google Maps work.
- The Project is programmed using Python3 and pygame library
- It provides the best shortest route from a user-specified source to a destination using 2 algorithms - Dijkstra and A*
- The blocks/walls/traffic can be set by the user or using functionalities added for making grids and arbitrary traffic.



GPS Routing:



Uses:-

- Compute Single Source Shortest Path.
- Highlight distances.
- Take account of traffic.



Major Functionalities

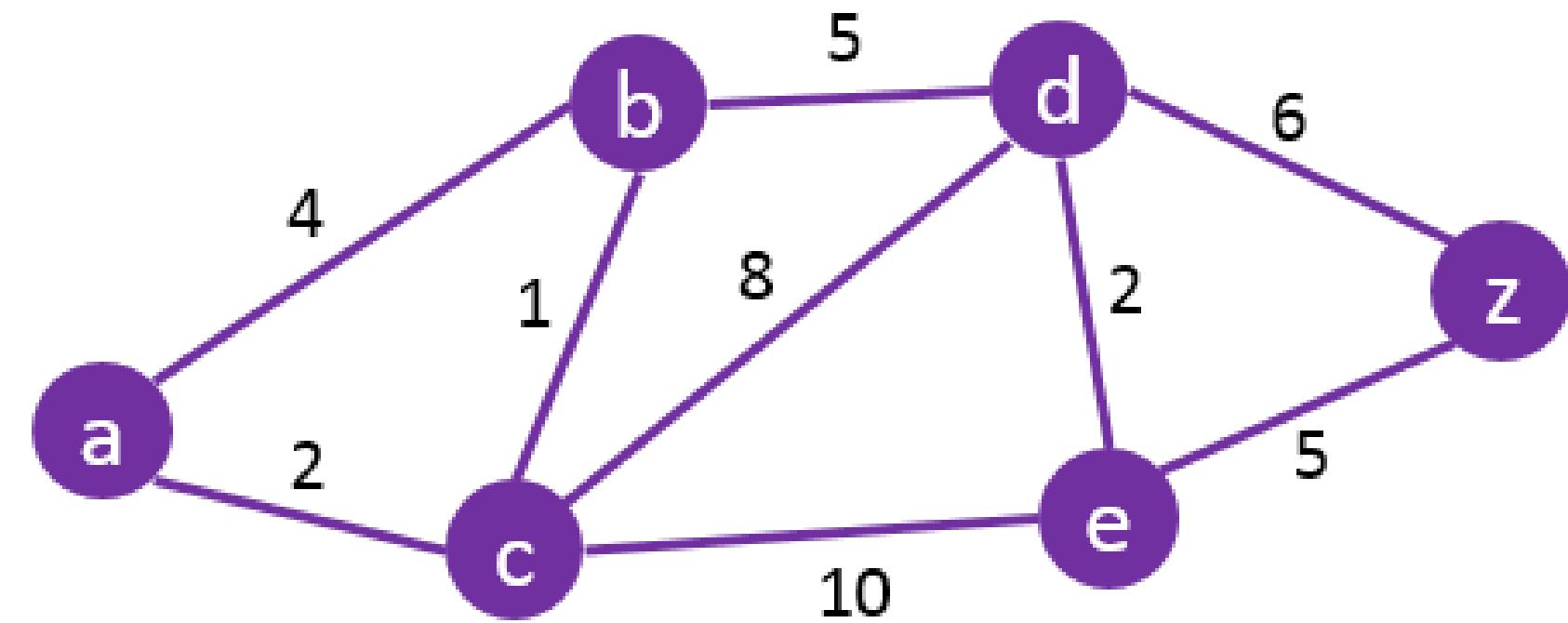
- Real Time Tracking
- Includes the Current Traffic Condition
- Accounts for the Muddy
- Generates random terrian
- Creates Random maze and solves



Prerequisites...

Algorithms:

- Djikstra algo
- A* Path algo



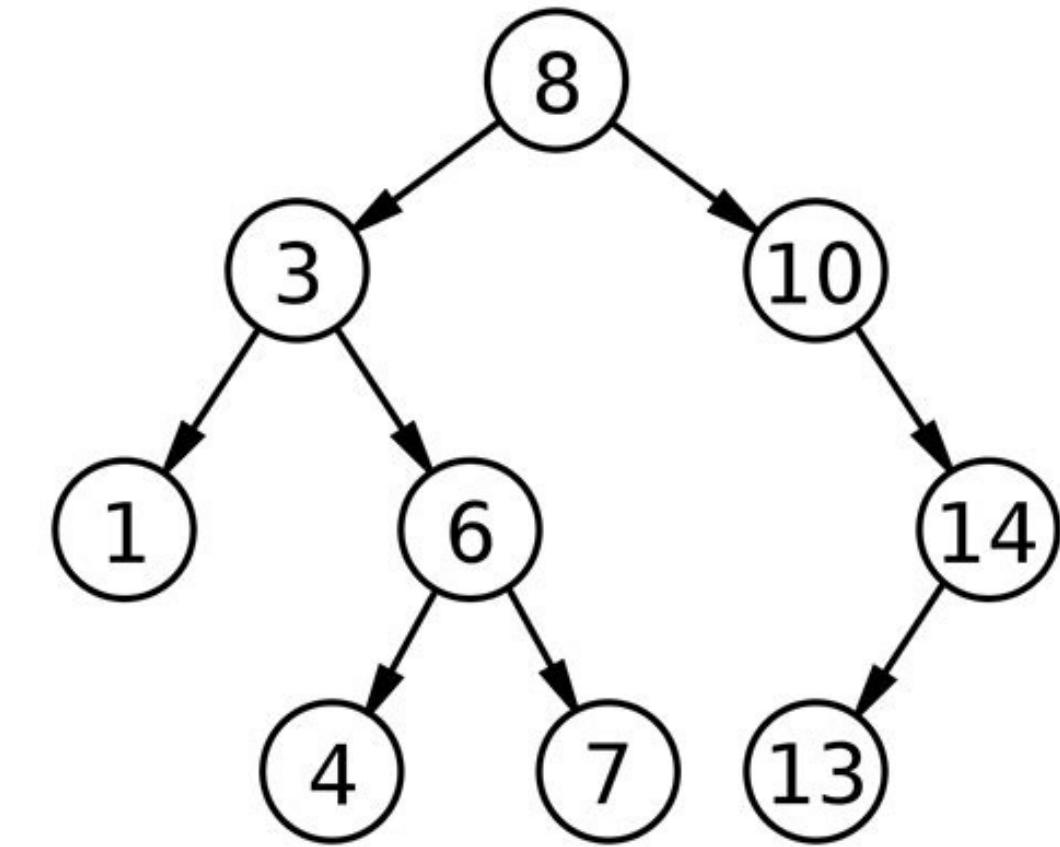
Dijkstra's Algorithm

What is the shortest path to travel from A to Z?

Djikstra basics...

- Used for finding the shortest paths from a source node to the destination node.
- It is a greedy algorithm that works on optimal move strategy

A* algo basics...



- A Search* (weighted): arguably the best pathfinding algorithm.
- Uses heuristic functions to guarantee the shortest path much faster than Dijkstra's Algorithm

Work distribution:-

- Aniket: Algorithms, All Utility Functions
- Adarsh: Button Class, Pygame, Random Mazes, Exception Handling
- Rajat : Node class, Visualisations, Random Terrain, File I/O

What we learnt in the project:



- Data structures:
 - Graphs
 - Queues
- Shortest Path Algorithms:
 - Dijkstra algo
 - A* algo
- OOPS using Python
- Python3 Libraries:
 - Pygame, Seaborn, Time, Random
- Collaborative Project Management Tools:
 - Atlassian Jira Software
 - Repl.it

Resources: -



I n s p i r a t i o n

clementmihailescu/ **Pathfinding-Visualizer**

A visualization tool for various pathfinding algorithms.



Contributor



Issues



1k
Stars



333
Forks



clementmihailescu/Pathfinding-Visualizer

A visualization tool for various pathfinding algorithms. - clementmihailescu/Pathfinding-Visualizer

GitHub

Thank you!



Hope you liked
our project and
we met your expectation!