

Route Planner

REVIEW

CODE REVIEW 6

HISTORY

Requires Changes

1 specification requires changes

Hi there,

I really appreciate your hard work and dedication towards this project. You had a very good and intuitive approach towards the solutions!! You have really tried very well and implemented the majority of the requirements.

You did a wonderful job in this submission, you nailed all the rubrics except just a few 💪

You just have to pass a few missing specifications of the project rubric to pass your project with flying colors. 😊

Your project needs to meet the following requirements to pass this project-

- Although the "euclidean distance" provided by you can be USD as a heuristic function, there are some other similar calculations, please kindly also provide them with comments n when you can use them also.
Hint: Like "manhattan distance" in the case when only 4 direction traversal is allowed and such.

For more details, please kindly go through my comments in the review section. They'll help you complete the remaining requirements.

Also, note that there is absolutely no limit on the number of submissions you can make on any project before it is accepted, so do not worry anything about that and patiently keep working on improving your submission by following all the provided feedback on how to meet the remaining requirements to pass the project!

As a helpful suggestion, for your continued learning in this amazing field of DSA and problem-solving, [here](#) is a highly resourceful and thorough set of important and thought-provoking problems that you might want to check out!

Also, as an added resource, for more in-depth knowledge on some advanced Data Structures and Algorithms, I would gladly like to point out [this](#) great compilation to you.

I look forward to seeing your skills continue to grow in Data Structures and Algorithms.

All the best.

Thank you 😊

Correctness

Running test.py shows "all tests pass".

Kudos! Your project passes all the test cases and corner quirks that can be thrown at it.

The student implements all required methods.

You have implemented all the required methods efficiently without any evident bug.

Well done!

The heuristic function used to estimate the distance between two intersections is guaranteed to return a distance which is less than or equal to the true path length between the intersections.

Although the "euclidean distance" provided by you can be used as a heuristic function, there are some other similar calculations, please kindly also provide them with comments when you can use them also.

Hint: Like "manhattan distance" in case when only 4 direction traversal is allowed and such.

Choice and Usage of Data Structures

Code avoids obvious inappropriate use of lists and takes advantage of the performance improvement afforded by sets / dictionaries where appropriate. For example, a data structure like the "open_set" on which membership checks are frequently performed (e.g. `if node in open_set`) should not be a list.

Your code uses the most optimal data structures like you have used in the shortest path finding methods and others which is providing the performance boost over the conventional data sets that you have avoided using for program efficiency in your code.

Well done on your choice of data structures!

This item is a judgement call. Student code doesn't need to be perfect but it should avoid big performance degrading issues like...

...unnecessary duplication of lists

...looping through a large set or dictionary when a single constant-time lookup is possible

🎉 **Congrats your project does not contain any unnecessary performance overhauls and most of the methods and functions in your code are optimal and efficient.**

Well done again!

🔍 RESUBMIT

📄 DOWNLOAD PROJECT

6

CODE REVIEW COMMENTS





Best practices for your project resubmission

Ben shares 5 helpful tips to get you through revising and resubmitting your project.

[▶ Watch Video \(3:01\)](#)

RETURN TO PATH

Rate this review

START