

Learning Outcomes

This project is a venue for the students to achieve the learning outcomes below:

- LO1.** Collaboratively build systems that consider a number of paths or strategies in order to improve its performance in achieving its goal in less amount of computing time, or by some other metric of performance.
- LO5.** Articulate ideas and present results in correct technical written and oral English

Students will demonstrate their skills in modeling, developing, and evaluating search algorithms of their choice. They are going to simulate how pathfinding is computed. They will also write a report explaining the strengths and weaknesses of their simulator.

Project Requirements

1. Students need to explore ONE (1) blind search algorithm and ONE (1) heuristic search algorithm they learned in class, and write a program in any programming language to simulate the optimal pathfinding for the eateries around DLSU as shown in Figure 1. Please note that this is an old map. The eateries may have relocated or may no longer exist. Consider at least 20 eateries.



Figure 1: DLSU Food Map

2. The chosen algorithm should be dynamic so that it is able to change its behavior if the goal state is changed or if the state space is changed, assuming that there will be a chance of a new eatery being established or an existing eatery being closed.

3. When searching for the best eatery, there are several heuristics (rules of thumb) that can help guide your decision. These heuristics can involve subjective factors, personal preferences, and external reviews. You can also ask ChatGPT about this.
4. Students must compare the differences in algorithms' performance with regard to the time complexity, memory complexity, and optimality.
5. Students need to apply their creativity in the program to interact with users and display the output in either text-based or graphical form.
6. Any references made to the existing codes must be acknowledged. Codes should not be 100% copied, however, they can be adopted and modified to suit your requirements. A serious penalty will be given to a plagiarized project.

Report Guideline

In addition to the above program, students must write a report containing the following sections:

Cover Page – DLSU Logo, Assessment Title, Name of Members and Submission Date.

I. Introduction

- Describe the problems and your proposed ideas to solve the problems.

II. Methodology

- Explain in detail the process that you used in translating the real-world information into formulas in the knowledge base.
- Explain the chosen algorithms. Use simple language to explain the approach, steps, or logic the algorithm follows.
- You may use illustrations to aid your description.

III. Results and Analysis

- Provide a screenshot of outputs with explanations.
- Write a comparison between the chosen algorithms.
- Summarize the strengths and limitations. Provide suggestions for the limitations (if any).

IV. References

- Cite all references including examples.
- Use IEEE citation format: <https://www.scribbr.com/category/ieee/>

V. Contributions of Each Member

- Explain in detail the contribution of each group member to the completion of the project.

The minimum number of pages for the report is 5, but keep in mind that you will be graded based on its substance, readability and completeness.

Deliverables

You are to submit the following through AnimoSpace on or before the deadline set by the instructor:

- The executable program (ready to run). This should be in a folder named **App**. If there are additional files needed for the program to run, make sure to include them. Include in this folder the instructions needed to run the program.
Include instructions on how to execute it, if it's not intuitive from the program itself like a README.txt file.
- The complete source codes used for the program. Put these in a folder named **Source**.
- A video file containing a demo (installation instructions, sample runs, analyses). Name this file **Demo.<file extension>**.
- A .pdf document containing the report. Name this file **Report.pdf**.

The executable file and the source codes are to be submitted as a single zip file with a filename of the following format: **MC01_<surname1>_<surname2>_<surname3>_<surname4>_<surname5>.zip**.

Please make sure that you comply with the naming conventions and format of the submission.

MC01 Rubric

Program	Program is executable and dynamic, meet the objective and creative. (5 marks)	Program is executable and dynamic, partially meet the objective, less creative. (4 marks)	Program is partially executable and partially meet the objective. (3 marks)	Some errors in the program (1-2 marks)	/ 5
Algorithm Understanding	High comprehension of the chosen algorithm's logic. (5 marks)	Medium comprehension of the chosen algorithm's logic. (4 marks)	Less comprehension of the chosen algorithm's logic. (3 marks)	Poor comprehension of the chosen algorithm's logic. (1-2 marks)	/ 5
Report	Report is completed with clear and sufficient explanations. (5 marks)	Report is less completed with sufficient explanations. (4 marks)	Report is less completed with less explanations. (3 marks)	Report is not completed. (1-2 marks)	/ 5
Presentation and Demonstration	Clear, confidence, show full understanding and able to answer questions. (5 marks)	Clear, confidence, show less understanding and able to answer questions. (4 marks)	Not clear, less confidence, show less understanding and having difficulties to answer questions. (3 marks)	Having difficulties to present and to answer questions. (1-2 marks)	/ 5
Total Marks					/ 20