

School of Computing Assessment Brief

Module title	Artificial Intelligence
Module code	COMP2611
Assignment title	Search Algorithms
Assignment type and description	Report detailing investigation, evaluation and customisation of algorithms (based on use of provided software).
Rationale	The assignment is designed to consolidate students' understanding of search algorithms and develop an appreciation of how different algorithm variants perform in relation to characteristics of the problem being addressed.
Length limit and guidance	8–10 pages (+ cover page). See submission instructions for more details.
Weighting	20% of total module grade
Submission deadline	Friday 21st February, 2025
Submission method	Gradescope (one member of pair submits then adds partner)
Feedback provision	Feedback will be provided via Gradescope.
Learning outcomes assessed	<ul style="list-style-type: none"> • Be able to employ AI techniques to solve problems • Evaluate the effectiveness of AI techniques. • Identify limitations of AI techniques.
Assignment Organiser	Brandon Bennett B.Bennett@leeds.ac.uk
Module Lead	Arash Rabbani: A.Rabbani@leeds.ac.uk

COMP2611 Coursework 1

Search Algorithms

Overview

Your task is to investigate the effectiveness of a variety of different search algorithms (and option settings) for solving: (A) sliding blocks puzzles; and (B) a simple task planning problem that might be assigned to a robot worker.

You will work on and submit your assignment as a *pair* (i.e. two students working together). You must both contribute to work on both parts of the assignment. ***It is not acceptable to work completely separately on different tasks*** (and this would be likely to result in lower marks as the parts are closely related.)

Provided you have kept up with the lectures and gone through the Search Exercises, I would expect that you can produce a report that will get you a pass mark will require both students to spend around 5 hours on this assignment. Of course, gaining a high grade is likely to take considerably longer.

Preliminaries

Your preliminary work for before starting this assignment should be to study the Search Exercise interactively executable notebook files that I have provided and can be accessed via a folder within the Learning Resources area of the module's Minerva pages.

Assignment Requirements and Marks

The overall structure of your assignment task is as follows:

A. Sliding Blocks Puzzle Search Investigation

This can be done working within the [Search Exercise 7 Colab notebook](#). You can use the notebook directly online, but ***take care to save your own code and results***.

Report sections and associated marks:

A1. Design a sequence of tests that show the strengths and weaknesses of different search algorithms when applied to sliding blocks puzzles. [10 marks]

This section should include the following:

- (a) Specification and explanation of the puzzle cases that you will test.
- (b) Specify and briefly explain any heuristic functions used in your tests. (These can be based on the ones given in Exercise 7, or you could design your own.)
- (c) Specify, in a clear and concise way, the complete set of tests upon which you will base the results presented in your answers to **A2** and **A3**.

A2. Present your results in the form of one or more tables. [6 marks]

A3. Give a list of key observations as concise bullet points. [6 marks]

B. Robot Worker Experimentation

Starting with the code given in the Search Exercise 5 notebook create and investigate your own robot worker scenario. A relatively small modification of the given scenario is sufficient. But to obtain a high grade you will need to introduce some significant novelty. Take care not to increase the complexity of the problem too much (If there are too many choices, the algorithms may fail to solve even apparently simple problems.)

Report sections and associated marks:

B1. Design and describe a robot worker scenario. [6 marks]

B2. Describe and implement a heuristic suitable for your scenario. [4 marks]

B3. Investigate the performance of a variety of search algorithms when applied to your robot worker scenario, and present results in table form. [4 marks]

B4. Give key findings as concise bullet points. [4 marks]

[40 marks total]

Read on for further details, suggestions and tips.

What and How to Investigate

In order to obtain high marks, you will need to carry out a thorough investigation of search algorithms within the limitations of the search algorithm software with which you have been provided. You should aim to cover the following different types of search algorithm and options, as well as different levels of difficulty of the problem itself:

- Different basic search algorithms
 - breadth-first,
 - depth-first, with fixed action choice ordering,
 - depth-first, with randomised action choice ordering,
 - best first,
 - A*.
- Different heuristics for the informed search algorithms (best first and A*). These can be based on the ones given in Exercise 7, or you could design your own. To obtain the most credit for your investigation heuristics you would need to devise heuristics that are applicable to the whole family of sliding block puzzles that can be formulated in terms of the `SlidingBlocksPuzzle` class, not just ones that work only for a particular problem (such as the simple ones given in the Exercise 7 Colab notebook).
- Whether or not loop checking is used.
- Puzzles of different levels of difficulty. (Think about what factors may affect ‘difficulty’.)

Note that the mark scheme for part **A1** allows for some flexibility in where you put your effort. You could concentrate more on conducting a wide range of test cases, or you could try to develop a good general heuristic for this kind of puzzle.

Tips

- When experimenting with different search configurations, proper testing will require adjustment of the *max nodes* parameter. Start with a small number such as 5000. If the search attempt terminates quickly without finding the solution you should raise the number to see if a solution can be found with a higher limit. If you still do not get a solution raise the limit further until, either you get a solution, or the algorithm runs for too long for you to wait. You can decide your own time limitations. Around 1 to 5 minutes would be a reasonable time to wait.
- Try problems of different difficulty by altering the value of the 8-puzzle initial state or the robot workers goal. I suggest you concentrate on 2 or 3 case, but you may have to try a few possibilities to identify ones that correspond to widely different levels of difficulty. Your hardest problem should preferably be one that is not solvable without using one of the heuristic search algorithms.

- Systematically record the results of your tests in tables. Each row should correspond to a particular search configuration and the columns should give the path length, total nodes tested and time taken. Also, indicate whether the search was successful. In the case where no solution has been found, there will be no path found and the number of nodes tested will be determined by the maximum node limit.

In order to present results clearly, you may wish to use a separate tables for different problem cases and/or different sets of values. However, it is up to you to judge what is the clearest and most informative way to present your results.

- Do **not** try to cram every possible test and result number you can think of into your report. Beyond giving a reasonable coverage of possibilities, you will not gain extra marks for sheer volume of results. You need to present your results in a clear way and could actually lose marks if your results tables are cluttered and difficult to understand.

See next page for submission instructions.

Submission Instructions

The submission for this assignment will be in the form of a document in **PDF** format, to be uploaded to **Gradescope**. Note that if you are working in a pair, *only one* of the pair will be submitting, but they must then add the other member of the pair after submission. The submission process is described in more detail in the template document that is provided (see below). Failure, to do this is likely to cause confusion in the grading process and a penalty of up to 5 marks may be deducted.

Your answer for each of the seven parts should be begin on a separate page (for ease of marking). It is expected that the answer for A1 would take up about 2 pages and the answers for the other parts would each fit on one page but you could go onto a second page if you feel you really need it (e.g. for results tables). But bear in mind that conciseness and clarity is a criteria in the assessment of all parts; so adding a lot of material that does not contain significant content will not increase your grade and could actually decrease it.

A PDF file illustrating the expected layout of your submission is provided and can be found the Assessment area of the module's Minerva pages. Note: this is just a guide for the layout, not an actual template file. You will need to create your own report file using the software of your choosing (e.g. Word or LaTeX/Overleaf). You will need to generate a PDF version of your file for upload to Gradescope.

When grading your report we will not be super-strict about the format, however if you depart from the suggested layout and your report is found to be unclear or difficult to grade then penalty marks will be deducted.

It is important that you *complete a declaration* at the beginning of your report. The possible declarations state either that you agree to work in a pair and get the same marks, or that you agree to work along but have your submission graded according to the same criteria as those working as a pair. The text of the declarations is given in the template file, so you just need to cut and paste the appropriate one from that document into the report. You also need to ensure that name and university email of the submitting student or students are included above the declaration. Failure to include one of the declarations at the beginning of your report is likely to result in a penalty of up to 5 marks being deducted.

Advice on Working in Pairs

Setting this assignment as one to be done in pairs is intended to give several benefits. Working in pairs should enable discussion and collaboration that will make the work more enjoyable, produce more interesting results and make doing the assignment less stressful.

As stated near the beginning of this document, it is expected that you work *together* for a significant proportion of the time you spend on this project. The different parts of the assignment are quite closely linked, so dividing it up and working separately is likely to significantly increase its difficulty and lead to poor results.

Especially for your initial work on each part (i.e. tasks **A1** and **B1**) I suggest you spend one or two hours working together to work out some ideas, do preliminary tests, and sketch out some parts of your answer. This can either face-to-face or via a video call (e.g. using Teams). The details of how you organise your collaboration are up to you, but I would recommend you create a shared document for your report using Sharepoint, Google Docs or Overleaf.

Once you have made some initial progress, it may then make sense to do some work separately on specific tasks. But be sure to keep in touch and have further meetings where you can integrate your work, evaluate what you have achieved and discuss what you should do next.