SIT215 Computational Intelligence
Assignment 1: Problem Solving

Grading and weighting: (% total mark for unit): 40%

Submission Deadline: 8:00 pm AEST, 13 April 2025 (Week 6 Sunday)

Overview and Learning Objectives

This assignment focuses on solving real-world problems using computational intelligence (CI) and practical problem-solving techniques. It encompasses various tasks for you to demonstrate your understanding of the knowledge discussed in the unit, like the basics of CI and soft computing, artificial intelligence agents, exploration of problem-solving strategies, and evaluation of solution quality. This assignment is designed to enable you to demonstrate your learning achievements in ULO1, ULO2, and ULO3 of this unit, including:

- Exhibiting proficiency in selecting and applying appropriate search algorithm(s) and data structures to act as an agent in problem resolution.
- Characterising and formulating navigation challenges utilising specific theoretical constructs and models.
- Developing, assessing, and critiquing the performance of various viable solutions to identify effective strategies.
- Communicating the design, methodology, and results of your investigative and developmental efforts clearly and comprehensively.

Problem to Solve

Many navigation programs, such as Google Maps, offer basic route planning but fail to consider accessibility features like kerb ramps, sidewalk widths, slopes, or indoor lifts. This assignment challenges you to design and implement a navigation solution tailored for wheelchair users. The primary foundation is demonstrating your problem-solving skills by applying key CI concepts and implementing the A* algorithm for wheelchair-friendly map navigation.

Submission

This is an individual assessment task. Please read this assignment tasksheet carefully. You **must** submit the following to the relevant Assignment Dropbox folder on the Unit Site no later than the given deadline:

1) **The main report – *[YourID]_assign1problemsolving_report.pdf*.** Your report should provide adequate information to effectively showcase your solution and learning outcome against the objectives in line with the assessment rubric provided. Every student, regardless of the target grade (P/C/D/HD), must use the standardised report structure, including *Cover Page, Problem Formation, Approach, Implementation Overview, Results and Evaluations, Conclusion, Acknowledgement (if any),* and *References* sections. Refer to Additional Guide for You to Complete this Assignment Effectively of this tasksheet for details.
2) **All codes were developed or used in this assessment.** Your code must include appropriate documentation (sufficient internal comments) explaining the logic behind your solution's key components. You should also provide instructions on how to execute the code in a README file.

Submit the following files as part of your submission:

- **Code** file: Provide your complete solution as a **Jupyter Notebook (.ipynb)** or **Python file (.py)**.
- **README** file: Provide a **README.txt** or **README.md** file with clear instructions on how to run your code, specifying any required libraries or dependencies.

  - Note: If your code includes the usage of any open-source libraries, you need to provide those libraries or make them accessible to the assessor (e.g., by providing a link to a download site and instructions on how to install and use the library in your solution).
- **Bonus Task 5 deliverables (If applicable)**: If you completed the bonus task, submit the GUI code, either within your main code file or as a standalone **.ipynb or .py** file. Additionally, submit a 1-minute narrated demonstration video **(**e.g., Using Deakin Panopto to create and upload your video or uploading your own local video file as [YourID]_bonus_task_demo**.mp4)**
  - Note: It is the student's responsibility to ensure the video is accessible.

Late Penalties

In accordance with Faculty assessment policies, late submissions to the submission folder will incur a penalty based on Deakin's Penalties for late submission and Due dates policies, https://www.deakin.edu.au/students/study-support/assessments-and-examinations/assessments

Your Tasks

If you aim for a High Distinction (HD), you must complete Tasks 1, 2, 3, and 4 for full evaluation. Task 5 (Bonus Task) is optional and can be attempted by any student regardless of their target grade to earn up to 2 additional marks.

Task 1: Environment creation and problem formation (P/C-level Criteria)

1.1) Create an environment indicating wheelchair-accessible routes and facilities in a chosen locale with at least 20 path segments in a selected area (e.g., campus, neighbourhood, etc).

- Employ mapping tools like Google Maps (Google My Maps: https://support.google.com/mymaps/), OpenStreetMap (https://www.openstreetmap.org/), or any relevant tool to annotate each accessible path segment.
- Add markers for key amenities or facilities like wheelchair-accessible restrooms, parking spaces, shops, and lifts.

Notes:

- The guide with 20 path segments is a reasonable startup scale for you to implement A* effectively in Task 2.
- You can gather environmental information using physical surveys, approximate sketches, online research, virtual street views, or collaboration with local disability organisations. If exact measurements are unavailable, you may use reasonable figurative estimations to represent environmental features clearly.

1.2) In your report – under the sections: *Problem Formation*, and *Approach*, include the following (but not limited to)

- Describe your defined problem and environment.
- Justify the features included and the tools used for map creation.

Task 2: Basic navigation implementation (P/C-level Criteria)

2.1) Implement the A* algorithm for your wheelchair map navigation.

- Represent the environment as an adjacency matrix to demonstrate connections between locations and associated costs (e.g., distance or travel time).

2.2) Test the implementation with different start and end points. Ensure the returned paths are valid and optimal.

2.3) In your report – under the sections: *Approach*, *Implementation Overview*, and *Results and Evaluations*, include the following (but not limited to)

- Explain your agent and heuristic choice used in your solution.
- Justify why your heuristic fits the environment.

- Explain the key components of your solution and how they work with annotated code. (It can be demonstrated via code comments in your code files or screenshot embedded in your report, with line numbers indicated clearly for effective assessment.)
- Explain the implementation outcome and testing results of your solution.

Task 3: Enhance environment and heuristics comparison (D-level Criteria)

3.1) Add at least 10 more path segments to expand the environment, involving a total of minimal 30 path segments.

3.2) Add additional environmental constraints or terrain features, for example, kerb ramps, slope, obstacles, etc.

3.3) Create a new heuristic for pathfinding that accounts for these environmental details/constraint features and compare it to the initial heuristic used in Task 2: Basic navigation implementation.

3.4) In your report – under the sections: *Approach*, *Implementation Overview*, and *Results and Evaluations*, include the following (but not limited to)

- Explain the new heuristic's logic and impact on your navigation results.
- Elaborate on how the deterministic knowledge (fixed constraints) inspires your design of the new heuristic.
- Compare the performance and accuracy of both heuristics. Use data or visualisations to support your analysis.

Task 4: Performance enhancement with an alternative algorithm (HD-level Criteria)

4.1) Implement an alternative pathfinding algorithm by your choice (e.g., Dijkstra's Algorithm, Depth-first-search, Contraction Hierarchies, or Transit Node Routing).

4.2) In your report – under the sections: *Approach*, *Implementation Overview*, and *Results and Evaluations*, include the following (but not limited to)

- Compare your implemented alternative pathfinding algorithm's performance with A* in terms of efficiency, accuracy, and scalability respectively.
- Based on your evaluation results, elaborate on their suitability for a wheelchair navigation problem under various conditions, such as varying constraints over time or large-scale maps.

Task 5 (Bonus): Graphical user interface (Bonus +2 marks, eligible for all P/C/D/HD goals)

5.1) Create a graphical user interface (GUI) in your program with all the following elements:

- enabling a user to input the enquiry path's start and end points,
- visualising the calculated paths and costs on the map,
- displaying additional information about the path and environment of your terrain, such as the estimated time to traverse it and any obstacles or landmarks that may be encountered along the way, and
- displaying a summary of the costs of each calculated route in pathfinding.

5.2) Produce a 1-minute narrated demonstration video showcasing the successful execution of your GUI implementation.

Marking

Your submission will be evaluated based on criteria that reflect the depth of your problem-solving skills, technical proficiency, theoretical knowledge, and communication effectiveness against the quality of your work. A numeric mark (out of a total mark of 40) will be determined by how well your work aligns with the grading criteria and its standard as outlined in the rubric available on the Unit Site under Resources>Assessment.

Additional Guide for You to Complete this Assignment Effectively

Preparatory Learning Activities

To complete this assessment task, you will need to first develop an understanding of a range of topics covered in this unit in weeks 1 to 4. Given the assessment deadline, this may require you to learn these topics independently to understand the scope as soon as possible in earlier weeks. The topics that you will need to be familiar with are:

- Agents and Environment
- Path Search Strategies
- Heuristic
- Solution Quality and Evaluation

Understanding these areas and how they inter-relate will make it far easier to understand the learning materials, and explain your investigations and outcomes in this assignment. Our advice is that you use this assignment as a basis for the study of these underlying areas to assist in integrating the knowledge covered in this unit into a meaningful 'whole', which supports completing this assessment task.

Report Structure

<table>
<tr><th>Sections</th><th>Guide for you</th></tr>
<tr><td>Cover Page</td><td>Must provide the following checklist to ensure clarity in your submission. Tick (✓) the boxes next to the tasks you have completed. Your submission will be assessed based on the tasks you mark as completed. Ensure each marked task is fully supported with corresponding sections in your report and code.
<table>
<tr><th>Tasks</th><th>Completed (✓)</th><th>Notes (Optional)</th></tr>
<tr><td>Task 1: Environment creation and problem formation (P/C)</td><td>☐</td><td></td></tr>
<tr><td>Task 2: Basic navigation implementation (P/C)</td><td>☐</td><td></td></tr>
<tr><td>Task 3: Enhance environment and heuristics comparison (D)</td><td>☐</td><td></td></tr>
<tr><td>Task 4: Performance enhancement with an alternative Algorithm (HD)</td><td>☐</td><td></td></tr>
<tr><td>Task 5: Graphical user interface (Bonus Task)</td><td>☐</td><td></td></tr>
</table></td></tr>
<tr><td>Problem Formation</td><td>It is about the problem definition in your own words. Show your map in this section. Hints: focus on the accessibility challenges and objectives for wheelchair navigation.</td></tr>
<tr><td>Approach</td><td>Explain your methodology and the algorithm(s) used to address the defined problem. Hints: Focus on CI concepts for route planning, such as environment, agents, search strategies, and the aspects that shaped your approach. Use appropriate terminology and examples from class discussions or practical exercises to justify your choices, and/or provide any evidence of your engagement with these topics, such as notes from peer discussions, screenshots from forums, or your contributions to the unit's Teams support channel.</td></tr>
<tr><td>Implementation Overview</td><td>Provide a clear narrative explaining the key functional components of your solution, ensuring it aligns with (but does not duplicate) your code comments. If including code snippets or screenshots, ensure line numbers are visible for clarity. Describe the code structure and logic, ensuring it aligns with the project goals. The explanation should be structured as follows:
Primarily, address Basic navigation implementation (P/C-level Criteria)
If applicable, include separate subsection(s) addressing Enhance environment and heuristics comparison (D-level Criteria) and/or those for HD-level Criteria</td></tr>
</table>

Results and Evaluations	Present the outcomes of your working solution, employing visual aids like graphs, charts, tables, figures, etc, to enrich your analysis. Discuss the significance of these results and what they indicate about your solution's performance and effectiveness. The explanation should be structured as follows: • Primarily, present the results and evaluation for (P/C-level Criteria) tasks • If applicable, provide separate subsection(s) elaborating relevant D- and/or HD-level task deliverables, respectively.
Conclusion	Summarise your key findings aligned with your implementation and results, highlighting how the deterministic environment shaped your CI approach and problem-solving strategies in the applied real-world scenarios.
Acknowledgement	Record any external assistance, if applicable.
References	List all citations and resources used in proper referencing style.

Expected Report Length

Your report should be concise yet comprehensive, focusing on the key components of your solution while maintaining clarity and depth. If completed correctly, *as a guide*, the expected length is approximately 8 to 14 single column A4 pages, using a font size of 11 with single line spacing, covering all essential sections as the outlined **Report Structure**:

- Cover Page: 1 page
- Problem Formation and Approach: 2-3 pages
- Implementation Overview: 2-4 pages with annotated code snippets
- Results and Evaluations: 2-4 pages, including visuals
- Conclusion, Acknowledgements and References: 1-2 pages

Getting Help and Support

Students are encouraged to collaborate by discussing tasks and assisting each other in understanding concepts, models, and algorithms. Engaging in class sessions (Lecture, Prac) and participating in the unit site discussion forum or Teams support channel can strengthen your understanding and contribute to peer learning. However, this is an individual assessment, and all development work and report writing must be your own. If using external references, ensure they are properly cited. Collusion and plagiarism will result in severe academic penalties. If you need clarification on completing this task, seek guidance from the teaching team.

Guidance Support

The teaching team is available to assist you with this task. If you need help understanding the material or requirements, seek support during class time or post your questions in the Discussion Forum on the Unit Site, where responses can also benefit your peers.

Report Writing Help

While the teaching team can offer guidance on report writing, ensure you review the instructions in this task sheet carefully. Additional support is available through the Writing Mentors program, which provides assistance with academic writing, including report preparation. For more information, visit http://www.deakin.edu.au/students/studying/study-support/writing-mentors.

Feedback

Students are actively encouraged to seek formative feedback from peers and teaching staff, on their work completed before the submission deadline, to ensure they are on track with this task. Feedback may be obtained during weekly scheduled practical classes upon request. Talk to us, and we'll support you! Besides, marking feedback will be provided after the evaluation process in 15 business days after the submission due date.

End of Assignment 1 Tasksheet.