

Problem Solving and Programming Exercise on Combinatorial Problems,
Computational Complexity and Search

1 Task description

- 1.1 Choose an optimisation problem, from the lectures, a real world problem, or from the literature.
- 1.2 Develop a new optimisation algorithm, or apply appropriate methods from the lectures to solve the problem.
- 1.3 Find the worst-case time complexity of the algorithm.
- 1.4 Write a report where you describe the problem, formulate the solution, and explain your choice of algorithm.

2 Some guidelines

- 2.1 You are free to choose any search or analytical method that you think would be suitable to solve this problem. However, you should give the rationale behind your decision, and demonstrate that the algorithm works.
- 2.2 If you choose to use a stochastic local search algorithm, you must define and implement all of its components (search space, solution set, neighbourhood relation, etc.). Please refer to the lecture notes for the complete list of components of SLS algorithms.
- 2.3 Please make sure that your report does not exceed 2,000 words. Your marker will stop marking at 2000 words.
- 2.4 Send the report and your code directly to aldeida.aleti@monash.edu. The assignment is due on Friday 5pm, Week 6. However, you can have until 9am Monday morning if you need it. If you need more time, send me an email asking for an extension, giving a reason, and giving a date you need an extension until.
- 2.5 Self-assess your work using the marking rubric and include it in your submission.

3 Marking rubric

Use this marking rubric to self- and peer-assess. Submit a highlighted self-assessed copy.

Fail

- Any evidence of academic integrity problems.
- Insufficient detail to understand the proposed solution.
- Substantial problems with clarity or referencing.

Pass

- Unambiguous rationale of the choice of search space.
- Appropriate choice of neighbourhood relation (optimisation algorithm).
- Correctness of the proposed solution.

Credit

- Persuasive research-based argument on the choice of search algorithm.
- Evidence of successful implementation of the algorithm.

Distinction

- Substantial use of scholarly literature (3-6 references).
- Evidence of critical reflection on the advantages and disadvantages of the selected search algorithm and its components.

High Distinction

- Some critique of the research literature.
 - Use of appropriate programming practices in the implementation of the algorithm.
 - Complexity analysis of the proposed algorithm.
-

■