

Total: /100

Bonus/Late submission:

-5 for each late day

Program write-up: /15

---

Should include: approach, comparison of two heuristics, etc. Easy to read and makes sense.

Should at least include a table of analysis with 100 cases.

Students do not need to test every depth. In general, they should have a good range (from 2 to 20).

Any experience you gained by doing this project. If it doesn't work, please describe the areas that you might be missing.

A proper output file is included, please refer to the sample output file for formatting purposes.

Program correctness: /75

---

-20 if the program can only solve simple puzzles with depth < 10, etc.

-10 if the program worked, but didn't test 100 cases.

-10 if the search cost is not about the same as stated on the website with the corresponding length

-5 if the program doesn't follow the specification. Each step of the solution needs to be printed out.

Give partial points if the program is not solving any puzzle. Depending on the quality of the code.

Say it has all major functions implemented (+40)

No major functions implemented (+20)

Program: /10

---

Efficiency. Organization of the program. Comments.

-5 if it takes too long to run (> 1 minute to solve a complex puzzle)

Deduct a few points if the programming style is too bad (e.g. one big main). The code is hard to read. The program doesn't follow the guideline.