The code base to start this assignment is found in the course git repository. You may need to pull the latest changes and resolve merge conflicts in a few Makefiles. The source code is in `cs4300-code-ai-agents/prog/VertexCover`, and builds `cs4300-code-ai/bin/00bin-o/VertexCoverSolver`.

View the `README.md` in the source directory for a brief tour of the Vertex Cover problem and the software that has already been created. The section on Tuning the Solver has suggestions of possible code edits to improve the solver’s performance.

**Requirements**

- Create an improved objective function.
- Consider improved neighbor definitions, implement if found.
- Generate multiple problems of various vertex counts. For example, 5, 10, 20, 40, 80, etc. Make several problems of each size to allow for statistical variation.
- Tune the simulated annealing temperature parameters for problems of size 80. These are the config parameters `start_temperature`, `min_temperature`, and `temperature_reduction`. Leaving resolution at 1000000 should be fine.
- Document your findings for simulated annealing tuning.
- Determine realistic upper limits for vertex counts for each of the algorithms: hill climbing, first choice and simulated annealing. Use approximately 60 seconds of CPU time as the time limit.
- Identify approximate values for the config parameters: `restart` and `generation_limit` to enforce the approximately 60 seconds limit. Note these values may be different depending on the algorithm type.
- Produce a table with the vertex count and configuration parameters.

**Passoff**

Submit your source code and problem instances by committing and pushing the repository. Also submit your documentation in Canvas, in the PDF format.