CS4343 Game Development Project
Phase 2 Prototype Progress Report

Team Members

- Tan Weijian Colin – Producer + Asset + QA
- Koh Meng Hui – Knowledge Map + AI
- Tan Zhen Yang – AI
- Low Xu Yan Jessica – Level + GUI
- Alicia Lee Jiayi – AI

1. The Game

- Single-player detective murder mystery solving game
- Only one difficulty level in prototype
- Story: Someone is murdered inside a room in a house, at a certain time, with a weapon by one of his 4 family members
- Goal: Interview the family members and find out who is the murderer, when was he killed, and which weapon was killed

2. Gameplay

- 3rd-person controller used to explore the house (W,A,S,D keys control movement)
- Use mouse to hover and click over a character/weapon to interact with the character/weapon
- Use mouse to click to open/close doors
- To ask a character a question, click on the character, and click on the questions, and the answers will pop out
- On the top-left hand corner, there are 2 additional features:
    - Knowledge Map: shows which rooms and weapons have been explored (still in progress)
    - Knowledge Timeline: shows the answers obtained in a chronological order based on the characters which are asked
- After enough information is obtained, one can solve the murder mystery

3. Technical Challenge: AI Question Tree

- Premise: Clues, the answers, are not hard-coded, but randomly generated and spread out amongst the 4 characters
- Answers are inserted into structured Question-and-answer tree
- All the possible activities by the characters are stored in MurderData.cs
- Person class stores the activities of each character for the 3 time periods
- GenerateTimeline.cs generates the story for 3 time periods: BeforeMurder, DuringMurder and AfterMurder
    1. Generate a murderer x from Suspects
    2. Access the data and give x a weapon related activity one hour before murder
    3. Give x a false activity to cover up for (2)
    4. Give a weapon related activity to another suspect.
    5. Give an alibi for (4)
    6. Fill the rest of the timeline with activities

- AI.cs collects all these information and builds them into a Tree of QnNodes to generate the questions and answers
- Some questions unlock some answers. Detective has to ask the correct question in order to find the correct answers.

4. Achievements / Difficulties

- Managed to come out with a Constrains-Satisfaction rule-based algorithm
- Debugging C# within and outside Unity
- Difficult to understand and use one another's code

5. Prototype: What does it prove or disprove

- Every run of the game generates a new mystery story
- Increases re-playability and difficulty
- Able to traverse the Question-Answer Tree when detective ask the correct question

6. Future

- to include difficulty levels:
  - Easy: 2 rooms, (more witnesses)
  - Medium: 4 rooms,
  - Hard: all (less witnesses)
- **Heuristics:** add heuristics and cost functions to the tree, using a priority queue or heap
  - number of witnesses/alibi for murderer
  - number of red herrings
  - how suspicious action is (whether it is weapon generated activity)
  - number of rooms
  - how far you are from solving the puzzle
  - increase time period
- Progress checker to check where you are in the game, and to give additional intelligent clues if necessary to help you solve the puzzle
- Improve the camera using ray-casting
- Simple testing to determine difficulty is anticipated and tweaked correctly
- Weapons will give useful information