Due 1/26/2024
Siddarth Maddhula/Joshua Levin

Lab 1 Technical Report for *Memory Lane*

You wake up and find yourself chained in a dungeon, and you have to escape from your mysterious captor speaking through an intercom. As you explore, you hear a mysterious voice in your head that convinces you he's friendly. You must solve puzzles to escape with your new "ally" in *Memory Lane*, a psychological thriller game inspired by the escape room genre. *Zero Escape*, *Saw*, and the *Escape Room* movies served as more specific inspirations.

The earliest and perhaps biggest problem we had to deal with came in the dungeon room. We wanted three things to happen in quick succession and trigger each other into happening: dropping the cheese causing the doberman to break its chain, the chain breaking off the wall revealing a hole in the wall, and the key to the exit being shown to be in the hole from the beginning. This proved very hard to implement because we didn't want to include the wall or chain as an object or container. Making the hole in the wall an object in the wall with the key in there seemed hard to implement because if the wall was always undescribed, you would never be able to reach the hole in the first place. We wrestled around for a while before finally settling on toggling the described/undescribed status for several objects. When the player drops a cheese (either small as intended or large for a health penalty), the key suddenly switches locations from the undescribed hole in the wall to the dungeon itself. This actual change is visualized for the player with text.

One major technical achievement in our game is the use of scenes for both narration and setting. We imported some code to make the scenes progress simply by providing any key input. The scenes were used for the opening sequence, the entrance to most rooms, as well as the finale.

We also wrote many custom verbs in order to make our puzzles work. For instance, for one of the puzzles, you have to erase a chalkboard to trigger an event. Without creating the erase action, this mini-puzzle simply wouldn't work. There are many other instances of new actions being made as well, such as digging. Digging ended up being created as a sort of composite action. We made it so the plots in the greenhouse were closed, opaque containers so the player wouldn't be able to see their contents. We had to make them locked too so that they wouldn't be able to just "open" the plots whenever they wanted. So what did we make the key to the plots? The shovel, obviously. The "dig" command was essentially equivalent to the "unlock [container] with shovel/open[container]" commands. This proved to be a really elegant solution as no one ordinarily would even try to do anything close to the actual untranslated commands, and would most likely go for the efficient way. If you see soil, your first instinct is to dig.

In order to get reading a notebook to work, we had to override the precoded "read" command to make reading the notebook show some of the internal text, rather than being a substitute for the look command. In other words, if you look at the book, it gives the description, but if you read the book, it tells you about what's inside of it. This was a nice touch which gave us the opportunity to add to the ambiance of the story and give some lore and foreshadowing towards the plot twist at the end.

Some items in the game would display their status as a container and even list the items inside, when their container status had to be hidden for the puzzle to work. For example, there is chalk hidden in the eraser in the classroom, but the description of the classroom would show the chalk hidden in the eraser which would spoil the mini-puzzle. Thus, we had to override some of the Inform7 functionality to only display the names of the items themselves and hide the items within.

Due 1/26/2024
Siddarth Maddhula/Joshua Levin

Another small technical problem we had to overcome was figuring out how to hide the presence of an object in a room until an event triggers it. This was a small but tricky thing that took some thinking to overcome. Our solution was just to create the item but not place it in the room until the event triggered, such as with the desk labeled 12.