Run make in the project directory, and make sure board.txt is present in the folder.

Demo 1 will cover different functionality as outlined in the specification

- Ignores -random-board in favour of load
- Tests input integer handling
- Skip basement placement when loading from save
- Building road through residence you own / don't own
- Save game to "test.txt"
- Attempts to improve residence with resources you don't own / is not yours
- Win function Check
  - Our win function checks if a builder has won directly after making a move, and does not wait until next
  - "Yes" to playing again
- Prints board, sets basements
- Rolls a few times
- Place geese on the same tile it already is, no builders to steal from (resources = 0)
- Prints status
- EOF character, game exits and saves to backup.sv

Demo 2 will cover the beginning of them from a randomly generated board. It will go through the process of building the initial basements. Initially, some of the following errors will be checked:
- Try to build basement that's adjacent to another residence
- Try to build basement at someone else's residence

The Demo will then go into the *Beginning of Turn* phase. The following functionalities for this phase are shown throughout the demo:
- Setting loaded dice
- Setting fair dice
- Invalid roll
- Help command

In terms of the *During the Turn* phase, a bulk of the commands and the possible errors will be shown in this demo. Residences, Next, Status, Trade (both with a response of yes and no), Build-road, and improvement are all tested. Some of the following invalid entries are also tested:

- Trade with invalid player
- Trade with invalid resource
- Trade when both players don't have enough resources
- Trade when one player doesn't have enough resources

- Trade with yourself
- Try to improve your own residence without enough resources
- Try to improve a residence that doesn't have an owner
- Try to improve a residence that is owned by someone else
- Try to build road without enough resources
- Try to build road that is not adjacent to your own residence

Finally, in the demo the first player will roll a 7 on their first turn. This is to test the Goose from the beginning. You'll notice it takes no resources from the players as none of them have 10 resources. Additionally, when Blue places the goose on a tile with another player, they cannot steal from them as there are no resources to steal.

**Demo will reach EOF, so game will be saved to backup.sv

Demo 3 Run using ./constructor -board demo3layout.txt < demo3.in

Demo 3 will cover the beginning of the game and extensively show the building & geese functionality. Throughout the demo it will also demonstrate how a Builder can gain multiple resources from more than one Tile in a concurrent roll.

In this demo we will be loading in a customized board. Then, after rolling the number 11 eight times, Yellow will demonstrate the legal moves of building a road, showing the following functionalities in the oder below:
- You can only build a road adjacent to the current Builder's residence or road
- You cannot build a road over an existing road
- You need one Heat and one Wifi resource

After building the roads, we will cover the legal functionalities of building a residence:
- You cannot build a residence adjacent to another residence
- You cannot build a residence where a residence already exists
- You cannot upgrade a residence that the current Builder doesn't own
- You cannot upgrade a residence that doesn't exist
- You must build a residence that is connected to an adjacent road owned by the current Builder
- You need different amount of resources to build/upgrade residences
- You cannot upgrade a residence that is already at the maximum type

Lasty, we will demonstrate the GEESE:
- The geese takes away resources from Yellow since it has > 10 resources
- The current builder (Blue) can choose from the builders on the tile. Notice, Blue is on the tile but cannot be chosen
- Blue randomly steals a resource from Red