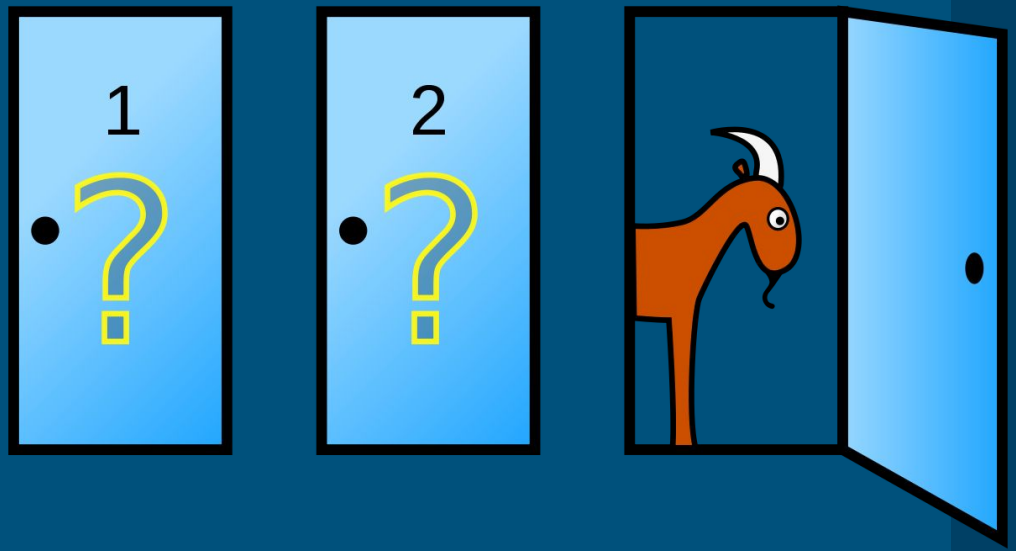
Final Project - Monty Hall Problem

WAP (Wireless Access Points) **Stars**:
Kyree Richardson, Trenton Paul, Christine Helenick

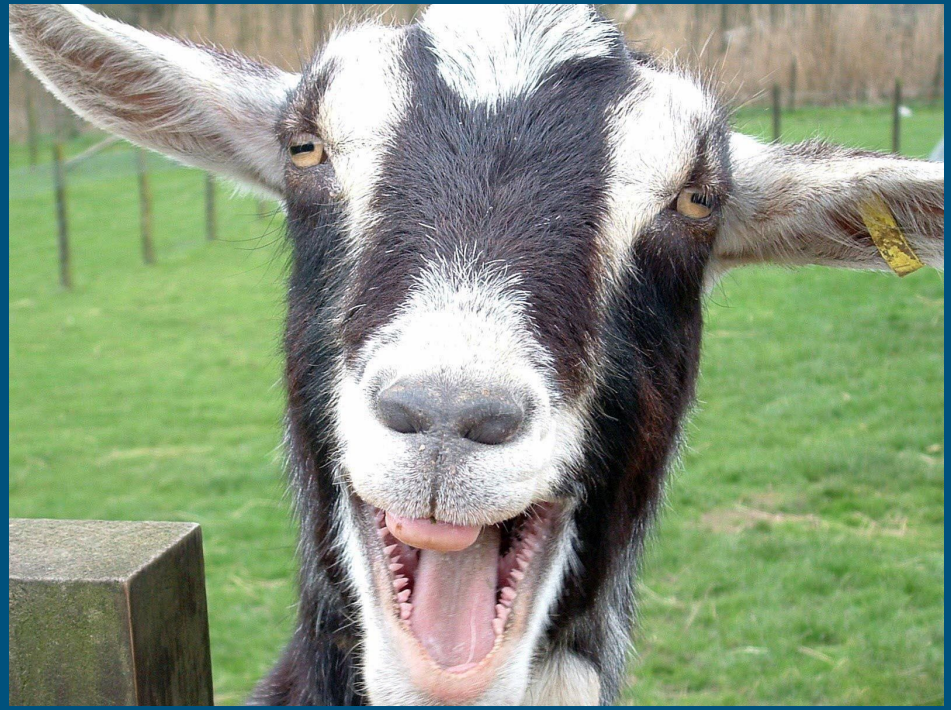
What is the Monty Hall Problem?



- You are shown 3 closed doors
- Select your door, but you don't get to see it yet
- A door with a goat is revealed = a losing door
- You don't know what's behind the 3rd door … do you want to swap doors or keep your original pick?

Our Approach

- Language choice: C++
- ASCII art / functions for open and closed doors to be printed during gameplay
- Create framework for game logic
- Add in the necessary client / server code
- Make sure the appropriate game code is divided into the correct spot -> client / server files



ASCII Art

- 4 functions for each scenario
  - All open
  - Door 1 is selected
  - Door 2 is selected
  - Door 3 is selected

```
void allDoors(){

    cout << R"(
 _____________   _____________   _____________
|\ _________ /|  |\ _________ /|  |\ _________ /|
| |  _ _ _ _  | |  | |  _ _ _ _  | |  | |  _ _ _ _  | |
| | | | | | | | |  | | | | | | | |  | | | | | | | |
| | |-+ 1 +-| | |  | | |-+ 2 +-| | |  | | |-+ 3 +-| | |
| | |-+ | +%| | |  | | |-+ | +%| | |  | | |-+ | +%| | |
| | |_|_|_|_| | |  | | |_|_|_|_| | |  | | |_|_|_|_| | |
| |    ___    | |  | |    ___    | |  | |    ___    | |
| |   [___] ()| |  | |   [___] ()| |  | |   [___] ()| |
| |         ||| |  | |         ||| |  | |         ||| |
| |         ()| |  | |         ()| |  | |         ()| |
| |           | |  | |           | |  | |           | |
| |           | |  | |           | |  | |           | |
| |           | |  | |           | |  | |           | |
|_|___________|_|  |_|___________|_|  |_|___________|_|

    )";
}

void door1(){

    cout << R"(
 _____________   _____________   _____________
|\ _________ /|  |\ _________ /|  |\ _________ /|
| |  /|,| |   | |  | |  _ _ _ _  | |  | |  _ _ _ _  | |
| | | 1 | |   | |  | | | | | | | |  | | | | | | | |
| | |,x,' |   | |  | | |-+ 2 +-| | |  | | |-+ 3 +-| | |
| | |,x   ,   | |  | | |-+ | +%| | |  | | |-+ | +%| | |
| | |/    |   | |  | | |_|_|_|_| | |  | | |_|_|_|_| | |
| |    /] ,   | |  | |    ___    | |  | |    ___    | |
| |   [/ ()   | |  | |   [___] ()| |  | |   [___] ()| |
| |       |   | |  | |         ||| |  | |         ||| |
| |       |   | |  | |         ()| |  | |         ()| |
| |       |   | |  | |           | |  | |           | |
| |      ,'   | |  | |           | |  | |           | |
| |    ,'     | |  | |           | |  | |           | |
|_|,'________|_|  |_|___________|_|  |_|___________|_|

    )";
}
```

Server

1. Connection
2. Receive guess
3. Send Goat
4. Receive new guess
5. Send correct / incorrect

Client

1. Establish connection
2. User guesses
3. Send guess
4. Receive goat
5. User switch?
6. Send new guess
7. Receive result

Thank you!

Possible goat pictures