

TESTING DOCUMENTATION

TESTED SITUATIONS

- Function to obtain port and host from the terminal was remove for easy of testing and debugging
- Connection was tested with different port and different host on the same router.
- Code was tested with preset secret key to ensure the computation of the program were equivalent to manual computations.
- Each element of the dictionary was printed at every step to see if the data was processed and stored as intended.
- One time was test both binary and decimal
- Code was test with different bids
- Test with two different computer on the same network
- Test with a user not familiar with code or programming
- Check were done to see if the right winners were selected for bids, challenges and invalid bids.
- Print messages at each interval to see if they were send as intended
- Test was done at the challenge phase to see if code work without a server bid on both server and client side.
- Server roll, Client roll, Server bid and Client bid were printed at every step to see if the information was stored properly.

LIMITATIONS

- If a character, letter or negative number is enter where there is a prompt for a number/ integer the program will not work properly.
- If the negative value are enter by the player then that player will win
- Code does that work when letters are entered for bid
- Code did not end of a certain amount of step as recommended. But code was implemented to have the users make higher bids resulting the conclusion of a lair dice round with the theoretically max bid of value 6 and frequency 10.
- Code was modified to exclude the args input from the command line to have code executed in the python shell and automatically run in Windows for ease of coding and testing. Program should work in terminal if code that commented out was re-entered and port and hostname entered manually from local and network testing is removed.