Relatório da disciplina Laboratório de Redes

Universidade Federal Rural do Rio de Janeiro

Semestre:2019-1

Aluno: Fabio Luiz Silva Nogueira

Matricula: 201278012-1

Sumário

[Descrição do Jogo 3](#_Toc13085753)

[Manual de Uso 3](#_Toc13085754)

[Diagrama de Sequência 4](#_Toc13085755)

[Código Fonte 5](#_Toc13085756)

[Funcionamento 8](#_Toc13085757)

[Referências 10](#_Toc13085758)

# Descrição do Jogo

O jogo trata-se de uma disputa de adivinhação entre dois jogadores. A adivinhação trata-se de um valor numérico aleatório entre 1 e 10. Para os jogadores disputarem no sistema, eles acessam a um servidor no qual fica armazenado os dados das partidas. Após a jogada, o servidor informa se o jogador acertou, caso não acerte, o próximo jogador realiza o seu palpite e assim sucessivamente, até alguém acertar. Além dos dois jogadores, o sistema contém um usuário Espectador, no qual pode acompanhar a partida sem interferir na mesma.

Após algum dos jogadores ganharem a partida, o servidor informa ao ganhador que ele ganhou, informa ao outro jogador que ele perdeu e informa para o espectador qual dos dois jogadores ganhou. Logo após o resultado, a conexão entre esses sistemas termina.

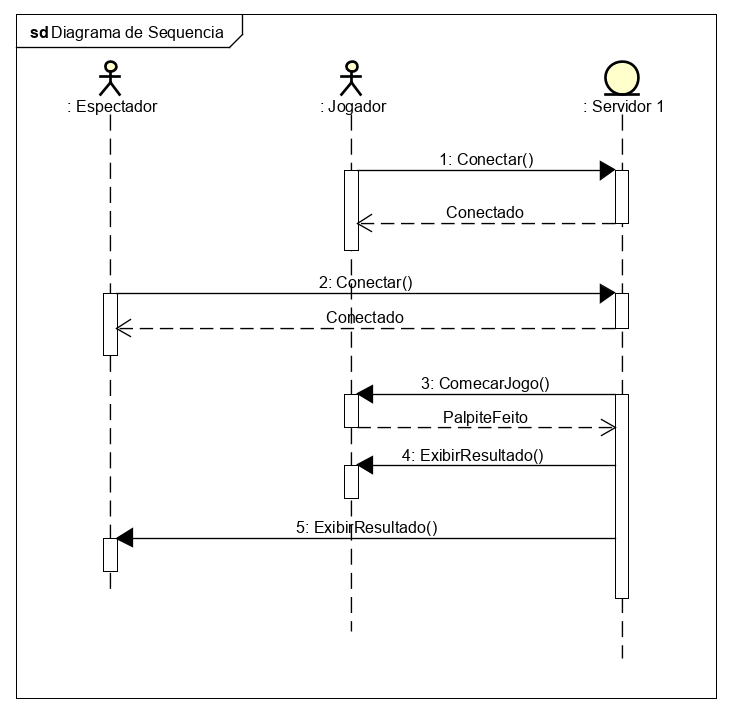
# Manual de Uso

Para poder utilizar o jogo, é preciso seguir algumas etapas à risca para que não ocorram erros. Seguem as etapas para a utilização de forma correta:

1. Servidor: Abrir em um terminal o arquivo “Servidor.py”
2. Jogador 1: Abrir em um terminal o arquivo “Cliente.py”
3. Jogador 2: Abrir em um terminal o arquivo “Cliente.py”
4. Espectador: Abrir em um terminal o arquivo “Espectador.py”

Logo após todo os usuários estiverem conectados, o servidor informará a todos que o jogo começou. O jogador só deverá realizar o palpite quando o servidor avisar que está na sua vez de jogar. No término do jogo, caso se deseje jogar novamente, deverá ser fechado todos os terminais e recomeçar o processo novamente.

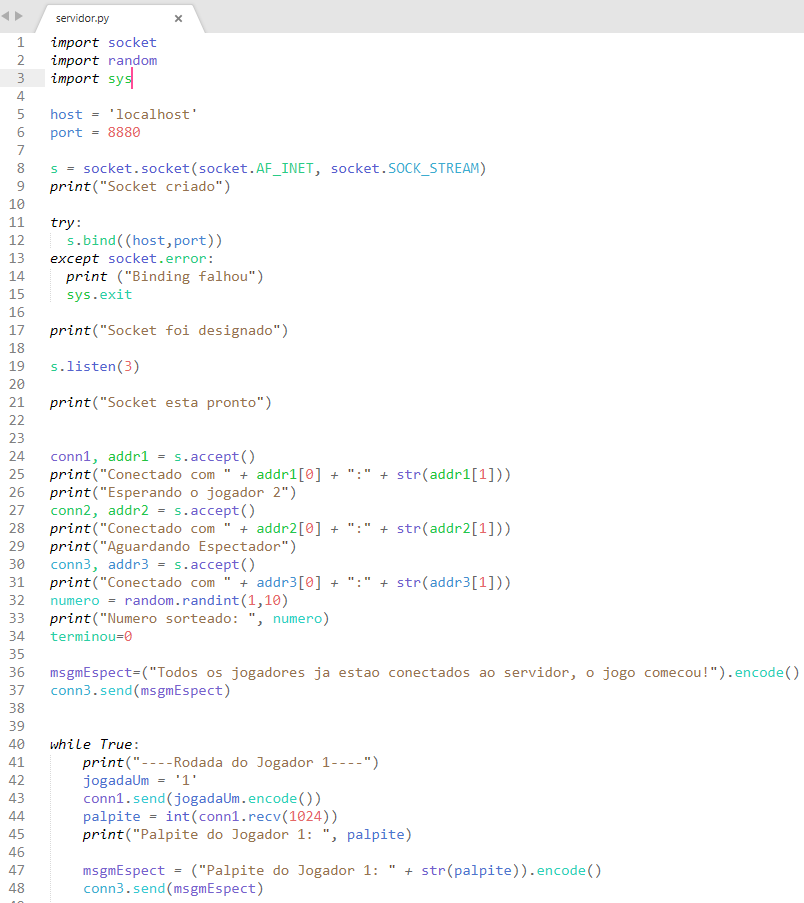
# Diagrama de Sequência



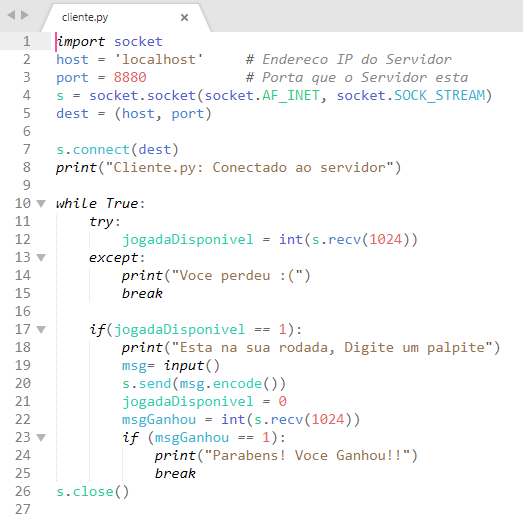
# Código Fonte

O jogo está dividido em três arquivos, Seguem eles:

**Servidor:**





**Cliente:**

**Espectador:**

# Funcionamento

O jogo foi implementado usando a linguagem Python 3 com o API de Sockets e protocolos TCP. Para seu funcionamento correto, o jogo foi separado em 3 arquivos diferentes.

Servidor

É o primeiro programa a ser executado. Ele funciona da seguinte forma:

* Importa as bibliotecas necessárias para sua execução (socket, random e sys). Escolhe um host e uma porta, no caso, localhost:8888.
* Cria um socket, e executa as funções sockets Bind e Listen.
* Armazena o Jogador 1 em conn1, Jogador 2 em conn2 e Espectador em conn3. Logo após informa o endereço e porta dos mesmos e que o jogo está pronto para começar
* Enquanto verdade, executa o código; Informa ao jogador 1 que está na vez dele através do envio do string ‘1’ e recebe de volta essa string e converte para inteiro. Se o palpite do jogador 1 estiver correto: Envia ao Jogador 1 que ele ganhou, que o 2 perdeu e ao espectador que o 1 ganhou.
* Caso o Jogador 1 não acerte o primeiro palpite, o algoritmo segue o mesmo raciocínio do passo anterior com o Jogador 2.
* O algoritmo só vai parar de executar quando algum dos dois jogadores ganharem e no caso, entrar em um dos “Ifs” e parar com o comando break.

Cliente

Logo após o servidor está sendo executado, dois Jogadores devem entrar no jogo, pelo processo cliente.py, que funciona da seguinte forma:

* Importa a biblioteca socket e informa o Host/Porta no qual o cliente deve se conectar, no caso de conectar-se com sucesso, é informado na tela
* Enquanto verdade: Tenta: se está ou não em sua rodada, prossegue com o código, caso dê erro (except), significa que o outro jogador ganhou e o atual perdeu.
* Se estivar na partida do mesmo: Informa que está na vez e pede um palpite, captura o palpite no teclado e transforma numa string a ser enviada ao servidor. Se o servidor retornar um int 1, significa que o mesmo acertou o palpite e o cliente notifica que ele ganhou e termina o código através do break e s.close().

Espectador

Logo após os clientes estarem conectados, o espectador se conecta ao servidor para assistir a partida, que funciona da seguinte forma:

* Importa a biblioteca socket e informa o Host/Porta no qual o espectador deve se conectar, no caso de conectar-se com sucesso, é informado na tela
* O servidor envia mensagens do que está ocorrendo para o espectador, que recebe através da variável msgm e o mesmo imprime essas mensagens em tela. Caso uma dessas mensagens seja de que algum dos jogadores ganhou, o programa imprime quem ganhou e termina sua execução através do break e s.close().

# Referências

Documentação de Sockets Python:

https://docs.python.org/3/library/socket.html

Guia Socket Básico do site Python Brasil:

https://wiki.python.org.br/SocketBasico

Video - How To Create a Multithreaded Server and Allow Many Clients - Python Networking Using TCP:

https://www.youtube.com/watch?v=ZwxTGGEx-1w