

UNIVERSIDADE FEDERAL DE ALAGOAS  
INSTITUTO DE COMPUTAÇÃO

PROFESSOR LEANDRO MELO DE SALES

# Redes de computadores

Relatório do projeto AB2

HECTOR SANTANA PINHEIRO  
AUDREY EMMELY RODRIGUES VASCONCELOS

[hsp@ic.ufal.br](mailto:hsp@ic.ufal.br)  
[aerv@ic.ufal.br](mailto:aerv@ic.ufal.br)

Maceió - AL  
Maio de 2021

# 1 Principais funcionalidades da aplicação

Pedra, papel e tesoura (do inglês Rock, paper, scissors), é um jogo de mãos recreativo e simples para duas pessoas, que não requer equipamentos nem habilidade.

Diferentemente dos métodos que se baseiam exclusivamente em sorte, pedra-papel-tesoura pode ser jogado com um pouco de habilidade. Principalmente se o jogo se estender por vários turnos com o mesmo jogador, este pode reconhecer e explorar a lógica do comportamento do adversário (perceber e anteceder as jogadas do adversário).

Para decidir quem ganha a partida são utilizadas as seguintes regras:

- Pedra ganha da Tesoura (a amassa e quebra)
- Tesoura ganha do Papel (o corta)
- Papel ganha da Pedra (a embrulha)

Na nossa implementação, a aplicação roda com um número par de clientes (2,4,6,8...), eles são combinados de dois em dois, pelo motivo do jogo ser jogado em dupla. Cada jogador escolhe uma das três opções (Pedra, papel ou tesoura) mostradas na tela apenas clicando em cima do nome da ferramenta que deseja utilizar. Após cada um fazer sua escolha, a aplicação decide o resultado com base nas regras citadas acima e divulga a decisão na tela de cada jogador, na tela do jogador vencedor aparece a mensagem de vitória e na tela do jogador perdedor aparece a mensagem de derrota, caso haja empate a mesma mensagem acusando o empate é mostrada na tela dos dois.

Nós criamos quatro arquivos para gerenciar o funcionamento da aplicação, são eles:

- client.py

Aqui a interface é inicializada, cada jogador pode iniciar o jogo e escolher qual será a sua jogada, e no final será mostrado o resultado para cada um.

- game.py

Neste arquivo nós recebemos a escolha do jogador e decidimos com base nas regras quem será o vencedor/perdedor da jogada, se não houver empate.

- network.py

É uma classe básica capaz de se conectar ao servidor e enviar/receber informações sobre o estado do jogador.

- server.py

Esse arquivo é responsável por lidar com todos os diferentes clientes que estão jogando nosso jogo. Ele armazenará e enviará informações a cada um dos clientes de forma adequada.

## 2 O que poderia ter sido implementado

Poderíamos ter feito a opção de jogar single player, onde o jogador entraria em confronto com a máquina.

Para efeitos de melhor visualização, poderíamos ter feito uma GUI com menu inicial tendo como opções "iniciar jogo multiplayer" e "manual de ajuda" para que o jogador pudesse entender como funciona o jogo se fosse preciso, e no caso de implementar a possibilidade de single player também teríamos a opção "iniciar jogo single player".

### 3 Dificuldades encontradas no desenvolvimento do projeto

Tivemos dificuldades no aprendizado e implementação da biblioteca pygame, por termos que criar uma interface mínima que o jogador pudesse interagir. Também tivemos um pouco de dificuldade na criação da thread, pois tivemos que fazer o encode e decode dos dados, porém ao pesquisarmos como fazê-lo apareciam versões antigas de como utilizar essas funções, que já não servem mais na versão 3.9.5 do python. Então passamos um bom tempo pesquisando como fazer encode e decode na versão atual da linguagem.