



**Universidade Federal do Ceará - UFC**

**Disciplina:** TI0145, Redes de Computadores I, 2018.1

**Professor:** Danielo Gonçalves Gomes

**Aluno(s):** \_\_\_\_\_

**Matricula:** \_\_\_\_\_

**Entrega:** 04/05/2018

### ATIVIDADES - LAB#3

Observações:

- ✗ A nota máxima em cada atividade de laboratório, será de 1 (um) ponto.
- ✗ A entrega das atividades seguirá o cronograma da disciplina
- ✗ A nota individual de cada aluno se dará mediante correção das questões e arguição no momento da entrega.
- ✗ Será atribuída nota **ZERO** em caso de utilização de sites com conteúdo impróprio, frases ofensivas ou conteúdo adulto.
- ✗ Serão aceitos arquivos **.py** ou **.ypnb** (python notebook), os códigos fonte deverão estar devidamente comentados e com a identificação dos componentes em sua primeira linha.

#### 1. Criando requisições via socket:

- a) Escolha um endereço de um site, crie um cliente **socket** e realize uma requisição HTTP de uma pagina web para exibir o conteúdo do site escolhido. Obs: Utilize o método **GET** ao enviar a requisição. Perguntas: O cliente utilizará TCP ou UDP, qual trecho do seu código indica isto? Na resposta retornada pela site, qual o **HTTP code** e o que ele significa?
- b) É de conhecimento que os serviços de rede utilizam portas as quais ficam aguardando alguma conexão. Com base nesta informação, escolha o endereço de um site qualquer e crie um **portScan** simples que irá varrer as portas de número 20,21,22,25,53,110,443,8080,3306 e caso encontre uma porta aberta (aguardando conexões), ele deverá exibir uma frase com o **número da porta** e o **NOME DO SERVIÇO** em que aquela porta por padrão pertence. Caso não seja uma porta de serviço padrão, ele deve apenas informar que a porta X esta aberta com um serviço desconhecido. Obs: Este aplicativo demora para rodar.

#### 2. Servidores e clientes TCP/UDP com sockets:

Digamos que exista um serviço de redes com a função de realizar saudações, onde um cliente se conecta e envia seu nome completo pela rede. O serviço de rede recebe o nome e devolve uma mensagem no formato: BOM DIA, BOA TARDE, BOA NOITE + SEU NOME dependendo da hora atual. Dica: Utilize a biblioteca **datetime** para capturar a hora e verifique o dia/tarde/noite com **if**.

- a) Desenvolva um **clienteUDP.py** e um **serverUDP.py** para o serviço utilizando sockets UDP que será executado no endereço **127.0.0.1** e **porta 5001**
- b) Desenvolva um **clienteTCP.py** e um **serverTCP.py** para o serviço utilizando sockets TCP que será executado no endereço **127.0.0.1** e **porta 5002**
- c) Quais as principais diferenças em termos de programação das abordagens nas letras a) e b)? Modifique seu **portScan** da questão 1 para incluir as portas TCP e UDP em que seu saudações server esta rodando, rode o server e o portScan no endereço 127.0.0.1. O que o portScan retorna?