

CLIENTE:

```
sock.connect((host, port))
print("Conectado ao servidor.")

# Solicita o nome do arquivo
arq = input('Digite o nome do arquivo: ')

# Converte o nome do arquivo para bytes e envia o comprimento e o nome
lenNameArq = len(arq.encode('utf-8')).to_bytes(2, 'big') #2 bytes do tamanho
msg = lenNameArq + arq.encode() # Concatena o comprimento e o nome do arquivo
sock.send(msg)

# Espera pela resposta do servidor
fileIsOk = int.from_bytes(sock.recv(2), 'big')
try:
    if fileIsOk == 0:
        tam = int.from_bytes(sock.recv(4), 'big') #recebe 4 bytes do tamanho do arq e converte para inteiro
        fd = open(DIRBASE+arq, 'wb')
        while tam > 0:
            recBytes = sock.recv(4096)
            fd.write(recBytes)
            tam -= len(recBytes) #atualiza o tamanho S
        fd.close()
        print("Arquivo recebido com sucesso! ")
    else:
        print('Arquivo inacessível')
```

O tamanho do nome do arquivo é calculado usando **len(arq.encode('utf-8'))** que terá armazenado na variável o número de bytes necessários para a string, que é convertido em um valor de 2 bytes no formato big-endian com **to_bytes(2, 'big')**. Logo abaixo é criado a variável **msg** que contem o tamanho e o nome concatenados em bytes, e após isso enviado ao servidor.

Em **int.from_bytes(sock.recv(4), 'big')** o cliente irá receber 4 bytes representando o tamanho do arquivo e será convertido para inteiro

SERVIDOR:

```
mensagem = con.recv(2) #recebe 2 bytes do tamanho
mensagem = int.from_bytes(mensagem, 'big') #converte para inteiro

# Abre o arquivo a servir ao cliente
fileName = con.recv(mensagem).decode('utf-8') #proximos bytes que contem o nome do arquivo
print ("Recebi pedido para o arquivo ", fileName)

fileName = DIRBASE+fileName
print(f"Caminho completo do arquivo: {fileName}")

if os.path.exists(fileName): #se encontrar o arquivo
    con.send(b'\x00\x00')
    fileSize = os.path.getsize(fileName) #pega o tamanho do arquivo em bytes
    con.send(fileSize.to_bytes(4, 'big')) #envia em 4 bytes em big endian

    fd = open (fileName, 'rb')
    print ("Enviando arquivo ", fileName)

    # Lê o conteúdo do arquivo a enviar ao cliente

    fileData = fd.read(4096) #4kb
    while fileData != b'':
        con.send(fileData)
        fileData = fd.read(4096)
    # Fecha o arquivo
    fd.close()
```

recebe os 2 bytes do tamanho do arquivo com **con.recv** e converte para inteiro usando **int.from_bytes(mensagem, 'big')**. No **con.recv(mensagem).decode('utf-8')** recebe o nome do arquivo decodificado em utf-8.

os.path.getsize(fileName) o servidor verifica o tamanho do arquivo, que é convertido em 4 bytes e enviado com: **con.send(fileSize.to_bytes(4, 'big'))** e enviado ao cliente