

```
lenNameArq = len(arq.encode('utf-8')).to_bytes(2, 'big')
msg = lenNameArq + arq.encode() # Concatena o comprimento
sock.send(msg)
```

Inicialmente no código do cliente o tamanho foi implementado usando o método len no nome do arquivo em formato de bytes no formato UTF-8, que é enviado ao servidor em 2 bytes e convertido para inteiro.

```
fileSize = os.path.getsize(fileName)
con.send(fileSize.to_bytes(4, 'big'))
```

Apos isso o servidor usa o getsize da biblioteca 'os' para obter o tamanho do arquivo armazenado no servidor e envia em 4 bytes no formato big-endian.

```
tam = int.from_bytes(sock.recv(4), 'big')
fd = open(DIRBASE+arq, 'wb')
while tam > 0:
    recBytes = sock.recv(4096)
    fd.write(recBytes)
    tam -= len(recBytes) #atualiza o
```

Voltando para o cliente que receberá esses 4 bytes e converterá para inteiro usando isso no laço de repetição para garantir que o arquivo inteiro seja recebido.