

1) Python e UDP

a)

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\vandi\OneDrive\Documentos\RedesII> & C:/Users/vandi/AppData/L
ocal/Programs/Python/Python312/python.exe "c:/Users/vandi/OneDrive/Docume
ntos/RedesII/Lista 1/server.py"
Atraso Min: 0.0
Atraso Max: 0.007609844207763672
Atraso Medio: 0.0002855195999145508
Atraso de Desvio Padrao: 0.000527048385247303
PS C:\Users\vandi\OneDrive\Documentos\RedesII> & C:/Users/vandi/AppData/L
ocal/Programs/Python/Python312/python.exe "c:/Users/vandi/OneDrive/Docume
ntos/RedesII/Lista 1/server.py"
█
```

b)

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

=> => writing image sha256:b589dd0984a9609e9e48a4e160b12be5cfe2e713b3ebc11d34ad24537dbce538
=> => naming to docker.io/library/lista1-client
[+] Running 2/2
✓ Container lista1-server-1 Created
✓ Container lista1-client-1 Created
Attaching to client-1, server-1
server-1 | Atraso Min: 0.00034427642822265625
server-1 | Atraso Max: 0.05919694900512695
server-1 | Atraso Medio: 0.0007134327888488769
server-1 | Atraso de Desvio Padrao: 0.002082209730907404
client-1 exited with code 0
server-1 exited with code 0
PS C:\Users\vandi\OneDrive\Documentos\RedesII\Lista 1> █
```

c) Se uma mensagem for perdida durante a transmissão usando o protocolo UDP, o programa prosseguirá sem interrupção, dado que o UDP não assegura a entrega de mensagens. Em um cenário onde um pacote se perde, ele não será reenviado nem haverá notificação de sua perda; o programa simplesmente continua operando sem reconhecimento desse evento. Portanto, em uma aplicação que envia 1000 mensagens, a perda de uma ou mais mensagens resultará em um número correspondente menor de mensagens recebidas e processadas pelo destinatário.