

Sistemas Distribuídos - CEFET – MG

Trabalho Prático 2

Transferência de Arquivos Peer-to-Peer

Aluno: Tarcísio Batista Prates

[GITHUB](#)

1. Teste 1 (Arquivo A – 10 KB, 2 Peers, Bloco 1 KB)

```
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Instale o PowerShell mais recente para obter novos recursos e aprimoramentos! https://aka.ms/PSWindows

PS C:\cloud\OneDrive - aluno.cefetmg.br\CEFET\MD_CEFET\Periodo_13\SD\TP1\pythonProject> python .\TP2\tp2_tarcisio.py --port 5000 --peers 1
27.0.0.1:5000 --file fileA.bin --seed
2025-11-14 21:39:05,441 [INFO] [Peer 5000] Inicializado
2025-11-14 21:39:05,441 [INFO] [Seeder] Arquivo: fileA.bin
2025-11-14 21:39:05,441 [INFO] [Seeder] Tamanho: 10240 bytes
2025-11-14 21:39:05,441 [INFO] [Seeder] Bloco: 10
2025-11-14 21:39:05,442 [INFO] [Seeder] SHA-256: 2c5ebb76320fe94ef2d2e83be56aaf634ecf3d48e406af6588541811edd548c
2025-11-14 21:39:05,442 [INFO] [Peer 5000] Seeder ativo. Aguardando peers...
2025-11-14 21:39:05,443 [INFO] [Peer 5000] Servidor ouvindo...
2025-11-14 21:39:08,799 [INFO] [Peer 5000] Enviou bloco 0 para ('127.0.0.1', 50788)
2025-11-14 21:39:08,824 [INFO] [Peer 5000] Enviou bloco 1 para ('127.0.0.1', 50781)
2025-11-14 21:39:08,825 [INFO] [Peer 5000] Enviou bloco 2 para ('127.0.0.1', 50782)
2025-11-14 21:39:08,840 [INFO] [Peer 5000] Enviou bloco 3 para ('127.0.0.1', 50783)
2025-11-14 21:39:08,841 [INFO] [Peer 5000] Enviou bloco 4 para ('127.0.0.1', 50784)
2025-11-14 21:39:08,842 [INFO] [Peer 5000] Enviou bloco 5 para ('127.0.0.1', 50785)
2025-11-14 21:39:08,856 [INFO] [Peer 5000] Enviou bloco 6 para ('127.0.0.1', 50786)
2025-11-14 21:39:08,857 [INFO] [Peer 5000] Enviou bloco 7 para ('127.0.0.1', 50787)
2025-11-14 21:39:08,857 [INFO] [Peer 5000] Enviou bloco 8 para ('127.0.0.1', 50788)
2025-11-14 21:39:08,858 [INFO] [Peer 5000] Enviou bloco 9 para ('127.0.0.1', 50789)

Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Instale o PowerShell mais recente para obter novos recursos e aprimoramentos! https://aka.ms/PSWindows

PS C:\cloud\OneDrive - aluno.cefetmg.br\CEFET\MD_CEFET\Periodo_13\SD\TP1\pythonProject> python .\TP2\tp2_tarcisio.py --port 5001 --peers 1
27.0.0.1:5000 --outfile recebido_A.bin
2025-11-14 21:39:08,792 [INFO] [Peer 5001] Inicializado
2025-11-14 21:39:08,794 [INFO] [Peer 5001] Servidor ouvindo...
2025-11-14 21:39:08,797 [INFO] [Peer 5001] Metadados recebidos de 127.0.0.1:5000
2025-11-14 21:39:08,798 [INFO] [Peer 5001] Progresso: 0/10 blocos
2025-11-14 21:39:08,799 [INFO] [Peer 5001] Recebeu bloco 0 de 127.0.0.1:5000
2025-11-14 21:39:08,824 [INFO] [Peer 5001] Recebeu bloco 1 de 127.0.0.1:5000
2025-11-14 21:39:08,826 [INFO] [Peer 5001] Recebeu bloco 2 de 127.0.0.1:5000
2025-11-14 21:39:08,840 [INFO] [Peer 5001] Recebeu bloco 3 de 127.0.0.1:5000
2025-11-14 21:39:08,841 [INFO] [Peer 5001] Recebeu bloco 4 de 127.0.0.1:5000
2025-11-14 21:39:08,842 [INFO] [Peer 5001] Recebeu bloco 5 de 127.0.0.1:5000
2025-11-14 21:39:08,856 [INFO] [Peer 5001] Recebeu bloco 6 de 127.0.0.1:5000
2025-11-14 21:39:08,857 [INFO] [Peer 5001] Recebeu bloco 7 de 127.0.0.1:5000
2025-11-14 21:39:08,857 [INFO] [Peer 5001] Recebeu bloco 8 de 127.0.0.1:5000
2025-11-14 21:39:08,858 [INFO] [Peer 5001] Recebeu bloco 9 de 127.0.0.1:5000
2025-11-14 21:39:09,790 [INFO] [Peer 5001] Todos os blocos recebidos!
2025-11-14 21:39:09,882 [INFO] [Peer 5001] Arquivo reconstruído em: recebido_A.bin
2025-11-14 21:39:09,822 [INFO] [OK] SHA-256 confere! Arquivo íntegro.
PS C:\cloud\OneDrive - aluno.cefetmg.br\CEFET\MD_CEFET\Periodo_13\SD\TP1\pythonProject> |
```

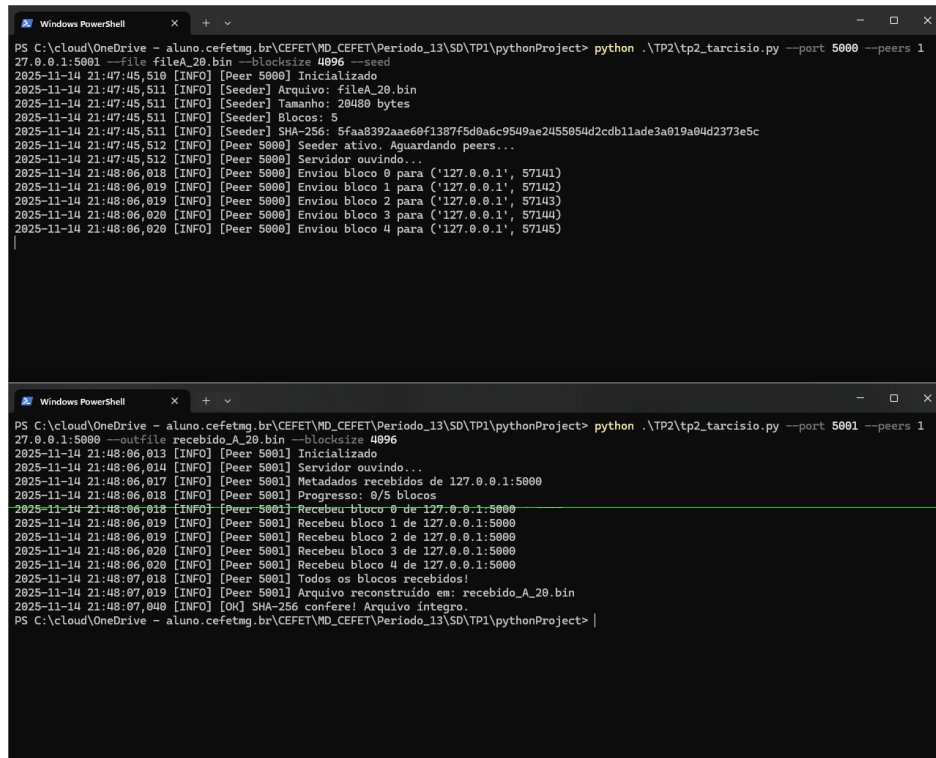
Figura 1: Terminal do Servidor em cima e abaixo do Cliente

No primeiro teste, utilizamos dois peers: o Peer 5000 como Seeder, contendo o arquivo de 10 KB, e o Peer 5001 como Leecher. O Seeder fragmentou corretamente o arquivo em 10 blocos de 1 KB e passou a aguardar conexões. Ao iniciar o Leecher, ele obteve os metadados do arquivo e iniciou a requisição dos blocos. Todos os blocos foram recebidos com sucesso e em sequência, como registrado nos logs. Após a transferência, o Leecher reconstruiu o arquivo e realizou a validação de integridade via SHA-256, que coincidiu com o hash do arquivo original.

O teste confirmou o funcionamento básico do protocolo P2P implementado, demonstrando que a fragmentação, transferência e remontagem ocorreram corretamente, sem perda de dados.

CMD	python .\TP2\tp2_tarcisio.py --port 5000 --peers 127.0.0.1:5001 --file fileB.bin - -blocksize 1024 --seed
CMD	python .\TP2\tp2_tarcisio.py --port 5001 --peers 127.0.0.1:5000 --outfile recebido_B2.bin --blocksize 4096

## 2. Teste 2 (Arquivo A – 20 KB, 2 Peers, Bloco 4 KB)



```
PS C:\cloud\OneDrive - aluno.cefetmg.br\CEFET\MD_CEFET\Periodo_13\SD\TP1\pythonProject> python .\TP2\tp2_tarcisio.py --port 5000 --peers 1
27.0.0.1:5001 --file fileA_20.bin --blocksize 4096 --seed
2025-11-14 21:47:45,510 [INFO] [Peer 5000] Inicializado
2025-11-14 21:47:45,511 [INFO] [Seeder] Arquivo: fileA_20.bin
2025-11-14 21:47:45,511 [INFO] [Seeder] Tamanho: 20480 bytes
2025-11-14 21:47:45,511 [INFO] [Seeder] Blocos: 5
2025-11-14 21:47:45,511 [INFO] [Seeder] SHA-256: 5faa8392aae60f1387f5d0a6c9549ae2455054d2c0b1ade3a019a84d2373e5c
2025-11-14 21:47:45,512 [INFO] [Peer 5000] Seedor ativo. Aguardando peers...
2025-11-14 21:47:45,512 [INFO] [Peer 5000] Servidor ouvindo...
2025-11-14 21:48:06,018 [INFO] [Peer 5000] Enviou bloco 0 para ('127.0.0.1', 57141)
2025-11-14 21:48:06,019 [INFO] [Peer 5000] Enviou bloco 1 para ('127.0.0.1', 57142)
2025-11-14 21:48:06,019 [INFO] [Peer 5000] Enviou bloco 2 para ('127.0.0.1', 57143)
2025-11-14 21:48:06,020 [INFO] [Peer 5000] Enviou bloco 3 para ('127.0.0.1', 57144)
2025-11-14 21:48:06,020 [INFO] [Peer 5000] Enviou bloco 4 para ('127.0.0.1', 57145)

PS C:\cloud\OneDrive - aluno.cefetmg.br\CEFET\MD_CEFET\Periodo_13\SD\TP1\pythonProject> python .\TP2\tp2_tarcisio.py --port 5001 --peers 1
27.0.0.1:5000 --outfile recebido_A_20.bin --blocksize 4096
2025-11-14 21:48:06,013 [INFO] [Peer 5001] Inicializado
2025-11-14 21:48:06,014 [INFO] [Peer 5001] Servidor ouvindo...
2025-11-14 21:48:06,017 [INFO] [Peer 5001] Metadados recebidos de 127.0.0.1:5000
2025-11-14 21:48:06,018 [INFO] [Peer 5001] Progresso: 0/5 blocos
2025-11-14 21:48:06,018 [INFO] [Peer 5001] Recebeu bloco 0 de 127.0.0.1:5000
2025-11-14 21:48:06,019 [INFO] [Peer 5001] Recebeu bloco 1 de 127.0.0.1:5000
2025-11-14 21:48:06,019 [INFO] [Peer 5001] Recebeu bloco 2 de 127.0.0.1:5000
2025-11-14 21:48:06,020 [INFO] [Peer 5001] Recebeu bloco 3 de 127.0.0.1:5000
2025-11-14 21:48:06,020 [INFO] [Peer 5001] Recebeu bloco 4 de 127.0.0.1:5000
2025-11-14 21:48:07,018 [INFO] [Peer 5001] Todos os blocos recebidos!
2025-11-14 21:48:07,019 [INFO] [Peer 5001] Arquivo reconstruido em: recebido_A_20.bin
2025-11-14 21:48:07,040 [INFO] [OK] SHA-256 confere! Arquivo íntegro.
PS C:\cloud\OneDrive - aluno.cefetmg.br\CEFET\MD_CEFET\Periodo_13\SD\TP1\pythonProject> |
```

Figura 2: Terminal do Servidor em cima e abaixo do Cliente

No segundo teste, avaliamos a transferência de um arquivo de 20 KB utilizando dois peers, agora com um tamanho de bloco maior (4096 bytes). O Seeder dividiu o arquivo em 5 blocos e aguardou conexões. O Leecher conectou-se ao Seeder, obteve os metadados e solicitou os blocos. Todos os 5 blocos foram transferidos rapidamente devido ao tamanho maior de cada bloco, reduzindo o número total de requisições. Após a recepção, o Leecher remontou o arquivo e confirmou a integridade, com o SHA-256 coincidente com o original.

O teste demonstrou que o sistema lida corretamente com blocos maiores e reduz o overhead de pedidos, mantendo a integridade e eficiência da transferência.

CMD	python .\TP2\tp2_tarcisio.py --port 5000 --peers 127.0.0.1:5001 --file fileA_20.bin --blocksize 4096 --seed
CMD	python .\TP2\tp2_tarcisio.py --port 5001 --peers 127.0.0.1:5000 --(outfile recebido_A2.bin) --blocksize 4096

### 3. Teste 3 (Arquivo B – 1 MB, 2 Peers, Bloco 1 KB)

```

Windows PowerShell
2025-11-14 21:51:58,176 [INFO] [Peer 5000] Enviou bloco 1001 para ('127.0.0.1', 54413)
2025-11-14 21:51:58,177 [INFO] [Peer 5000] Enviou bloco 1002 para ('127.0.0.1', 54414)
2025-11-14 21:51:58,196 [INFO] [Peer 5000] Enviou bloco 1003 para ('127.0.0.1', 54415)
2025-11-14 21:51:58,197 [INFO] [Peer 5000] Enviou bloco 1004 para ('127.0.0.1', 54416)
2025-11-14 21:51:58,198 [INFO] [Peer 5000] Enviou bloco 1005 para ('127.0.0.1', 54417)
2025-11-14 21:51:58,199 [INFO] [Peer 5000] Enviou bloco 1006 para ('127.0.0.1', 54418)
2025-11-14 21:51:58,200 [INFO] [Peer 5000] Enviou bloco 1007 para ('127.0.0.1', 54419)
2025-11-14 21:51:58,200 [INFO] [Peer 5000] Enviou bloco 1008 para ('127.0.0.1', 54420)
2025-11-14 21:51:58,201 [INFO] [Peer 5000] Enviou bloco 1009 para ('127.0.0.1', 54421)
2025-11-14 21:51:58,202 [INFO] [Peer 5000] Enviou bloco 1010 para ('127.0.0.1', 54422)
2025-11-14 21:51:58,202 [INFO] [Peer 5000] Enviou bloco 1011 para ('127.0.0.1', 54423)
2025-11-14 21:51:58,203 [INFO] [Peer 5000] Enviou bloco 1012 para ('127.0.0.1', 54424)
2025-11-14 21:51:58,203 [INFO] [Peer 5000] Enviou bloco 1013 para ('127.0.0.1', 54425)
2025-11-14 21:51:58,204 [INFO] [Peer 5000] Enviou bloco 1014 para ('127.0.0.1', 54426)
2025-11-14 21:51:58,204 [INFO] [Peer 5000] Enviou bloco 1015 para ('127.0.0.1', 54427)
2025-11-14 21:51:58,205 [INFO] [Peer 5000] Enviou bloco 1016 para ('127.0.0.1', 54428)
2025-11-14 21:51:58,205 [INFO] [Peer 5000] Enviou bloco 1017 para ('127.0.0.1', 54429)
2025-11-14 21:51:58,206 [INFO] [Peer 5000] Enviou bloco 1018 para ('127.0.0.1', 54430)
2025-11-14 21:51:58,206 [INFO] [Peer 5000] Enviou bloco 1019 para ('127.0.0.1', 54431)
2025-11-14 21:51:58,227 [INFO] [Peer 5000] Enviou bloco 1020 para ('127.0.0.1', 54432)
2025-11-14 21:51:58,243 [INFO] [Peer 5000] Enviou bloco 1021 para ('127.0.0.1', 54433)
2025-11-14 21:51:58,258 [INFO] [Peer 5000] Enviou bloco 1022 para ('127.0.0.1', 54434)
2025-11-14 21:51:58,260 [INFO] [Peer 5000] Enviou bloco 1023 para ('127.0.0.1', 54435)

Windows PowerShell
2025-11-14 21:51:58,197 [INFO] [Peer 5001] Recebeu bloco 1004 de 127.0.0.1:5000
2025-11-14 21:51:58,198 [INFO] [Peer 5001] Recebeu bloco 1005 de 127.0.0.1:5000
2025-11-14 21:51:58,199 [INFO] [Peer 5001] Recebeu bloco 1006 de 127.0.0.1:5000
2025-11-14 21:51:58,200 [INFO] [Peer 5001] Recebeu bloco 1007 de 127.0.0.1:5000
2025-11-14 21:51:58,200 [INFO] [Peer 5001] Recebeu bloco 1008 de 127.0.0.1:5000
2025-11-14 21:51:58,201 [INFO] [Peer 5001] Recebeu bloco 1009 de 127.0.0.1:5000
2025-11-14 21:51:58,202 [INFO] [Peer 5001] Recebeu bloco 1010 de 127.0.0.1:5000
2025-11-14 21:51:58,202 [INFO] [Peer 5001] Recebeu bloco 1011 de 127.0.0.1:5000
2025-11-14 21:51:58,203 [INFO] [Peer 5001] Recebeu bloco 1012 de 127.0.0.1:5000
2025-11-14 21:51:58,203 [INFO] [Peer 5001] Recebeu bloco 1013 de 127.0.0.1:5000
2025-11-14 21:51:58,204 [INFO] [Peer 5001] Recebeu bloco 1014 de 127.0.0.1:5000
2025-11-14 21:51:58,205 [INFO] [Peer 5001] Recebeu bloco 1015 de 127.0.0.1:5000
2025-11-14 21:51:58,205 [INFO] [Peer 5001] Recebeu bloco 1016 de 127.0.0.1:5000
2025-11-14 21:51:58,205 [INFO] [Peer 5001] Recebeu bloco 1017 de 127.0.0.1:5000
2025-11-14 21:51:58,206 [INFO] [Peer 5001] Recebeu bloco 1018 de 127.0.0.1:5000
2025-11-14 21:51:58,206 [INFO] [Peer 5001] Recebeu bloco 1019 de 127.0.0.1:5000
2025-11-14 21:51:58,227 [INFO] [Peer 5001] Recebeu bloco 1020 de 127.0.0.1:5000
2025-11-14 21:51:58,243 [INFO] [Peer 5001] Recebeu bloco 1021 de 127.0.0.1:5000
2025-11-14 21:51:58,258 [INFO] [Peer 5001] Recebeu bloco 1022 de 127.0.0.1:5000
2025-11-14 21:51:58,260 [INFO] [Peer 5001] Recebeu bloco 1023 de 127.0.0.1:5000
2025-11-14 21:51:58,392 [INFO] [Peer 5001] Todos os blocos recebidos!
2025-11-14 21:51:58,396 [INFO] [Peer 5001] Arquivo reconstruido em: recebido_B.bin
2025-11-14 21:51:58,409 [INFO] [OK] SHA-256 confere! Arquivo íntegro.
PS C:\cloud\OneDrive - aluno.cefetmg.br\CEFET\MD_CEFET\Periodo_13\SD\TP1\pythonProject>
  
```

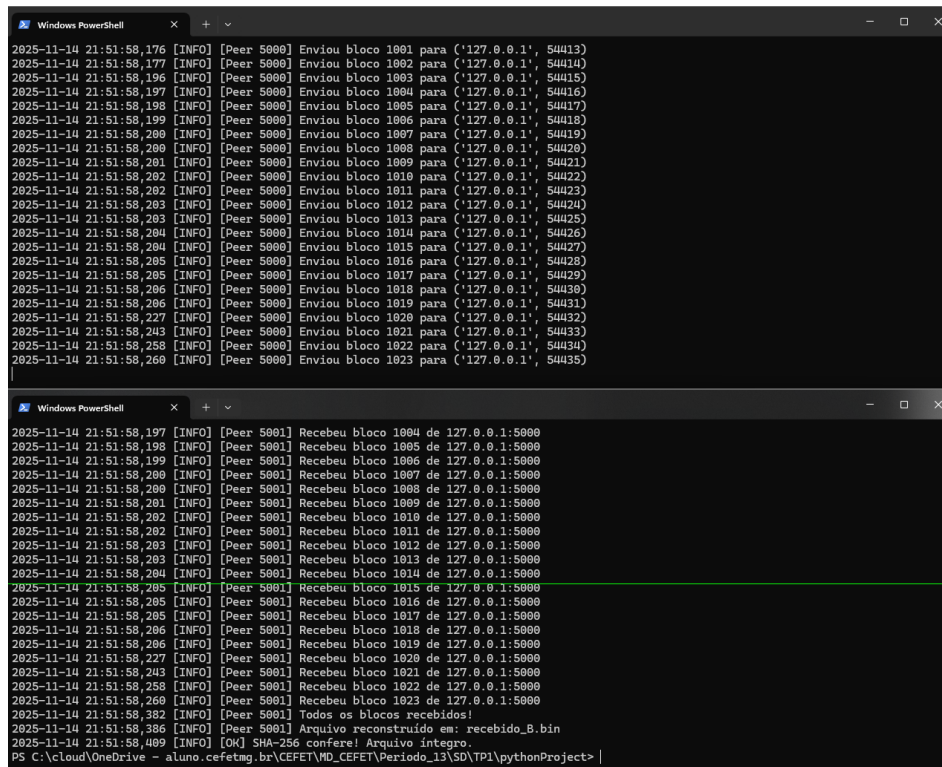
Figura 3: Terminal do Servidor em cima e abaixo do Cliente

No terceiro teste, realizamos a transferência de um arquivo médio de 1 MB, dividido pelo Seeder em 1024 blocos de 1 KB. Devido ao maior número de blocos, o volume de mensagens enviadas entre os peers aumentou significativamente, como evidenciado nos logs. O Leecher conectou-se ao Seeder, recebeu corretamente os metadados e iniciou a solicitação sequencial dos blocos. Todos os blocos foram recebidos sem falhas, e a reconstrução final resultou em um arquivo idêntico ao original, confirmado pela validação do hash SHA-256.

Este teste demonstrou que o protocolo permanece estável mesmo com um número elevado de blocos e maior carga de transferência, mantendo a integridade e completude do arquivo.

CMD	python .\TP2\tp2_tarcisio.py --port 5000 --peers 127.0.0.1:5001 --file fileB.bin - -blocksize 1024 --seed
CMD	python .\TP2\tp2_tarcisio.py --port 5001 --peers 127.0.0.1:5000 --outfile recebido_B.bin --blocksize 1024

#### 4. Teste 4 (Arquivo B2 – 1 MB, 2 Peers, Bloco 4 KB)



```
2025-11-14 21:51:58,176 [INFO] [Peer 5000] Enviou bloco 1001 para ('127.0.0.1', 54413)
2025-11-14 21:51:58,177 [INFO] [Peer 5000] Enviou bloco 1002 para ('127.0.0.1', 54414)
2025-11-14 21:51:58,196 [INFO] [Peer 5000] Enviou bloco 1003 para ('127.0.0.1', 54415)
2025-11-14 21:51:58,197 [INFO] [Peer 5000] Enviou bloco 1004 para ('127.0.0.1', 54416)
2025-11-14 21:51:58,198 [INFO] [Peer 5000] Enviou bloco 1005 para ('127.0.0.1', 54417)
2025-11-14 21:51:58,199 [INFO] [Peer 5000] Enviou bloco 1006 para ('127.0.0.1', 54418)
2025-11-14 21:51:58,200 [INFO] [Peer 5000] Enviou bloco 1007 para ('127.0.0.1', 54419)
2025-11-14 21:51:58,200 [INFO] [Peer 5000] Enviou bloco 1008 para ('127.0.0.1', 54420)
2025-11-14 21:51:58,201 [INFO] [Peer 5000] Enviou bloco 1009 para ('127.0.0.1', 54421)
2025-11-14 21:51:58,202 [INFO] [Peer 5000] Enviou bloco 1010 para ('127.0.0.1', 54422)
2025-11-14 21:51:58,202 [INFO] [Peer 5000] Enviou bloco 1011 para ('127.0.0.1', 54423)
2025-11-14 21:51:58,203 [INFO] [Peer 5000] Enviou bloco 1012 para ('127.0.0.1', 54424)
2025-11-14 21:51:58,203 [INFO] [Peer 5000] Enviou bloco 1013 para ('127.0.0.1', 54425)
2025-11-14 21:51:58,204 [INFO] [Peer 5000] Enviou bloco 1014 para ('127.0.0.1', 54426)
2025-11-14 21:51:58,204 [INFO] [Peer 5000] Enviou bloco 1015 para ('127.0.0.1', 54427)
2025-11-14 21:51:58,205 [INFO] [Peer 5000] Enviou bloco 1016 para ('127.0.0.1', 54428)
2025-11-14 21:51:58,205 [INFO] [Peer 5000] Enviou bloco 1017 para ('127.0.0.1', 54429)
2025-11-14 21:51:58,206 [INFO] [Peer 5000] Enviou bloco 1018 para ('127.0.0.1', 54430)
2025-11-14 21:51:58,206 [INFO] [Peer 5000] Enviou bloco 1019 para ('127.0.0.1', 54431)
2025-11-14 21:51:58,227 [INFO] [Peer 5000] Enviou bloco 1020 para ('127.0.0.1', 54432)
2025-11-14 21:51:58,243 [INFO] [Peer 5000] Enviou bloco 1021 para ('127.0.0.1', 54433)
2025-11-14 21:51:58,258 [INFO] [Peer 5000] Enviou bloco 1022 para ('127.0.0.1', 54434)
2025-11-14 21:51:58,260 [INFO] [Peer 5000] Enviou bloco 1023 para ('127.0.0.1', 54435)

2025-11-14 21:51:58,197 [INFO] [Peer 5001] Recebeu bloco 1004 de 127.0.0.1:5000
2025-11-14 21:51:58,198 [INFO] [Peer 5001] Recebeu bloco 1005 de 127.0.0.1:5000
2025-11-14 21:51:58,199 [INFO] [Peer 5001] Recebeu bloco 1006 de 127.0.0.1:5000
2025-11-14 21:51:58,200 [INFO] [Peer 5001] Recebeu bloco 1007 de 127.0.0.1:5000
2025-11-14 21:51:58,200 [INFO] [Peer 5001] Recebeu bloco 1008 de 127.0.0.1:5000
2025-11-14 21:51:58,201 [INFO] [Peer 5001] Recebeu bloco 1009 de 127.0.0.1:5000
2025-11-14 21:51:58,202 [INFO] [Peer 5001] Recebeu bloco 1010 de 127.0.0.1:5000
2025-11-14 21:51:58,202 [INFO] [Peer 5001] Recebeu bloco 1011 de 127.0.0.1:5000
2025-11-14 21:51:58,203 [INFO] [Peer 5001] Recebeu bloco 1012 de 127.0.0.1:5000
2025-11-14 21:51:58,203 [INFO] [Peer 5001] Recebeu bloco 1013 de 127.0.0.1:5000
2025-11-14 21:51:58,204 [INFO] [Peer 5001] Recebeu bloco 1014 de 127.0.0.1:5000
2025-11-14 21:51:58,205 [INFO] [Peer 5001] Recebeu bloco 1015 de 127.0.0.1:5000
2025-11-14 21:51:58,205 [INFO] [Peer 5001] Recebeu bloco 1016 de 127.0.0.1:5000
2025-11-14 21:51:58,205 [INFO] [Peer 5001] Recebeu bloco 1017 de 127.0.0.1:5000
2025-11-14 21:51:58,206 [INFO] [Peer 5001] Recebeu bloco 1018 de 127.0.0.1:5000
2025-11-14 21:51:58,206 [INFO] [Peer 5001] Recebeu bloco 1019 de 127.0.0.1:5000
2025-11-14 21:51:58,227 [INFO] [Peer 5001] Recebeu bloco 1020 de 127.0.0.1:5000
2025-11-14 21:51:58,243 [INFO] [Peer 5001] Recebeu bloco 1021 de 127.0.0.1:5000
2025-11-14 21:51:58,258 [INFO] [Peer 5001] Recebeu bloco 1022 de 127.0.0.1:5000
2025-11-14 21:51:58,260 [INFO] [Peer 5001] Recebeu bloco 1023 de 127.0.0.1:5000
2025-11-14 21:51:58,386 [INFO] [Peer 5001] Todos os blocos recebidos!
2025-11-14 21:51:58,409 [INFO] [OK] SHA-256 confere! Arquivo íntegro.
PS C:\cloud\OneDrive - aluno.cefetmg.br\CEFET\MD_CEFET\Periodo_13\SD\TP1\pythonProject> |
```

Figura 4: Terminal do Servidor em cima e abaixo do Cliente

Esta execução, o arquivo de 1 MB foi dividido pelo Seeder em blocos de 4096 bytes, totalizando 256 blocos. O Leecher conectou-se ao Seeder, recebeu os metadados e iniciou a solicitação dos blocos. Os blocos foram enviados pelo Seeder e recebidos sem falhas, com registro contínuo de transferências nos logs. Após receber todos os blocos, o Leecher reconstruiu o arquivo e confirmou a integridade utilizando o hash SHA-256, que correspondeu ao valor calculado pelo Seeder.

O teste confirma que o sistema funciona corretamente com blocos maiores e um número moderado de blocos, mantendo a integridade do arquivo após a reconstrução.

CMD	python .\TP2\tp2_tarcisio.py --port 5000 --peers 127.0.0.1:5001 --file fileB.bin - -blocksize 4096 --seed
CMD	python .\TP2\tp2_tarcisio.py --port 5001 --peers 127.0.0.1:5000 --outfile recebido_B2.bin --blocksize 4096

## 5. Teste 5 (Arquivo C – 10 MB, 4 Peers, Bloco 1 KB)

The image displays four terminal windows. The top two windows represent the server's perspective, showing outgoing connections and the distribution of file blocks to various peers. The bottom two windows represent the clients' perspective, showing incoming connections and the receipt of file blocks from the server and other peers. The logs are detailed, showing timestamps, peer IDs, and specific block numbers and sizes.

Figura 5: Terminal do Servidor em cima e abaixo do Cliente

Neste teste, utilizamos quatro peers para transferir um arquivo de 10 MB fragmentado em blocos de 4096 bytes, totalizando aproximadamente 2500 blocos. O Peer 5000 operou como Seeder, enquanto os peers 5001, 5002 e 5003 atuaram como Leechers conectados entre si. A distribuição dos blocos ocorreu tanto a partir do Seeder quanto entre os próprios Leechers, evidenciado nos logs: cada peer recebeu blocos de múltiplas fontes, o que confirma o comportamento P2P esperado. O volume de mensagens foi alto devido ao tamanho do arquivo e ao número de peers ativos.

Durante a execução, foi encerrado manualmente um dos terminais (um dos Leechers). Quando isso ocorre, os outros peers continuam requisitando blocos das conexões restantes. O peer que foi finalizado deixa de responder, mas os demais continuam a transferência normalmente, desde que exista pelo menos um peer ativo com os blocos restantes. Esse comportamento mostra que a rede não depende de um único Leecher e que a tolerância a falhas é limitada ao fato de que o Seeder ainda está disponível. Os peers que permaneceram ativos completaram a recepção dos blocos, reconstruíram o arquivo e confirmaram a integridade por meio do hash SHA-256, que coincidiu com o valor original. Isso demonstra que a interrupção de um peer não impede a conclusão da transferência pelos demais.

CMD	python .\TP2\tp2_tarcisio.py --port 5000 --peers 127.0.0.1:5001,127.0.0.1:5002 --file fileC.bin --seed
CMD	python .\TP2\tp2_tarcisio.py --port 5001 --peers 127.0.0.1:5000,127.0.0.1:5002 --outfile C1.bin
CMD	python .\TP2\tp2_tarcisio.py --port 5002 --peers 127.0.0.1:5000,127.0.0.1:5001,127.0.0.1:5003 --outfile C2.bin
CMD	python .\TP2\tp2_tarcisio.py --port 5003 --peers 127.0.0.1:5002,127.0.0.1:5001 --outfile C3.bin