

CENTRO UNIVERSITÁRIO FEI
ANTONIO GUSTAVO MUNIZ DA SILVA

22.119.001 - 0

HENRIQUE VITAL CARVALHO

22.119.078- 8

JOÃO VITOR DIAS DOS SANTOS

22.119.006-9

RELATÓRIO

ATIVIDADE 04 – Comunicação entre processos

Sistemas Distribuídos - CC37261
Prof. Ricardo Destro
Ciência da Computação
Noturno

São Bernardo do Campo

2022

SUMÁRIO

1. Teste: 5 processos 1 em paralelo	3
1.1 Média	3
1.2 Script.....	3
1.3 Logs.....	3
2. Teste: 10 processos 1 em paralelo	4
2.1 Média	4
2.2 Script.....	4
2.3 Logs.....	5
3. Teste: Maximo de chaves em 5 Segundos.....	6
3.1 Média	6
3.2 Script.....	6
3.3 Logs.....	7
4. TESTE: Tempo para gerar 100 chaves dados códigos iniciais próximos.....	8
4.1. Gráfico	8
4.1. SCRIPT	8
4.1. LOGS.....	8
5. Repositório do github	9

1. TESTE: 5 PROCESSOS 1 EM PARALELO

Para realizar esse teste utilizamos um script em bash que gerava 5 processos do tipo 1 simultaneamente, além disso também subimos um processo do tipo 2 e 3.

O resultado dessa execução é visto nos tópicos abaixo:

1.1 MÉDIA

Teste 1	
	0.851301193
	0.671792984
	0.697621822
	0.718264103
	0.788012743
	0.745398569

1.2 SCRIPT

Os valores usados de código inicial e N podem ser vistos no código abaixo:

```
script.sh
1  #!/bin/bash
2  #
3  python3 process_1.py 1000001 15000 >> output1.txt &
4  python3 process_1.py 1000002 15000 >> output2.txt &
5  python3 process_1.py 1000003 15000 >> output3.txt &
6  python3 process_1.py 1000004 15000 >> output4.txt &
7  python3 process_1.py 1000005 15000 >> output5.txt &
```

1.3 LOGS

Os logs da execução desses processos podem ser vistos no link abaixo:

<https://github.com/henriquevital00/multi-thread/tree/main/Logs/Teste%201>

2. TESTE: 10 PROCESSOS 1 EM PARALELO

Para realizar esse teste utilizamos um script em bash que gerava 10 processos do tipo 1 simultaneamente, além disso também subimos um processo do tipo 2 e 3.

O resultado dessa execução é visto nos tópicos abaixo:

2.1 MÉDIA

Teste 2	
	0.693898916
	0.722731113
	0.692470074
	0.846444607
	0.733022928
	0.741278648
	0.682179213
	0.673962116
	0.694933414
	0.731560946
	0.721248198

2.2 SCRIPT

Os valores usados de código inicial e N podem ser vistos no código abaixo:

```

script.sh
1  #!/bin/bash
2  #
3  python3 process_1.py 1000001 15000 >> output1.txt &
4  python3 process_1.py 1000002 15000 >> output2.txt &
5  python3 process_1.py 1000003 15000 >> output3.txt &
6  python3 process_1.py 1000004 15000 >> output4.txt &
7  python3 process_1.py 1000005 15000 >> output5.txt &
8  python3 process_1.py 1000006 15000 >> output6.txt &
9  python3 process_1.py 1000007 15000 >> output7.txt &
10 python3 process_1.py 1000008 15000 >> output8.txt &
11 python3 process_1.py 1000009 15000 >> output9.txt &
12 python3 process_1.py 1000010 15000 >> output10.txt

```

2.3 LOGS

Os logs da execução desses processos podem ser vistos no link abaixo:

<https://github.com/henriquevital00/multi-thread/tree/main/Logs/Teste%202>

3. TESTE: MAXIMO DE CHAVES EM 5 SEGUNDOS

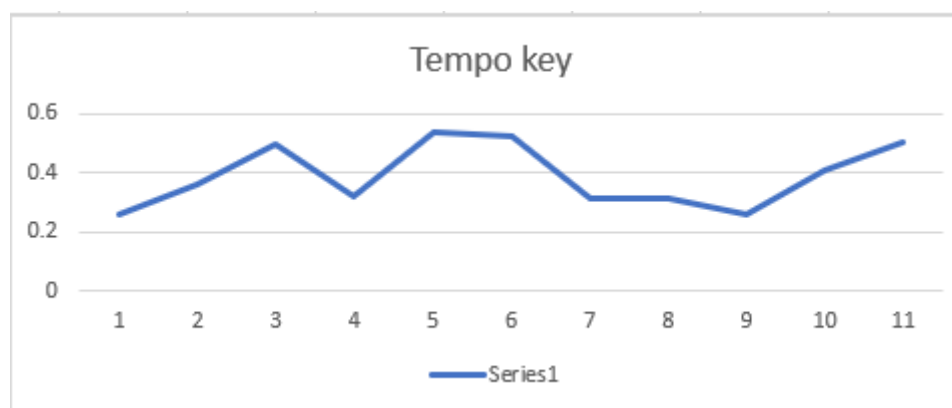
Para realizar esse teste utilizamos um script em bash que gerava N processos do tipo 1 no intervalo de 5 segundos, além disso também subimos um processo do tipo 2 e 3.

O resultado dessa execução é visto nos tópicos abaixo:

3.1 MÉDIA

Não é possível colocar aqui a tabela com todas as execuções, pois 11 chaves foram criadas nesse teste. Abaixo temos um gráfico das execuções

Teste 3	
	0.259267569
	0.364722013
	0.498720407
	0.320756197
	0.536353588
	0.522492409
	0.312213898
	0.317024231
	0.260898113
	0.410312414
	0.503122568
	0.69227767



3.2 SCRIPT

Os valores usados de código inicial e N foram gerados de forma randomica por meio do algoritmo abaixo:

```
time_test.py
1  import random
2  from ProcessOne import ProcessOne
3  import time
4
5  # five minutes
6  t_end = time.time() + 5
7
8  while time.time() < t_end:
9      n = random.randint(5000, 15000)
10     code = random.randint(10000000, 20000000)
11     process = ProcessOne(str(code), str(n))
12     process.send_message()
13     process.wait_callback()
14
```

3.3 LOGS

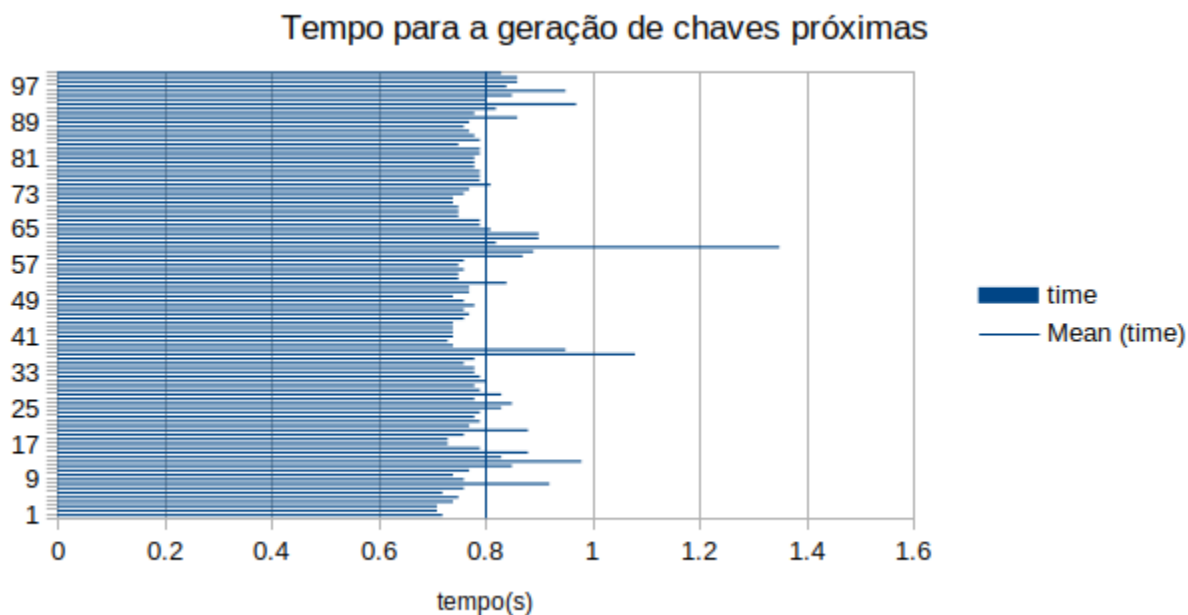
Os logs da execução desses processos podem ser vistos no link abaixo:

<https://github.com/henriquevital00/multi-thread/tree/main/Logs/Teste%203>

4. TESTE: TEMPO PARA GERAR 100 CHAVES DADOS CÓDIGOS INICIAIS PRÓXIMOS

Para realizar esse teste utilizamos um script que itera sobre 100 entradas pré-definidas e chama o processo 1 passando cada uma delas. O resultado dessa execução é visto nos tópicos abaixo:

4.1. GRÁFICO



4.1. SCRIPT

```
#!/bin/bash
#
tests=('10000001,15000' '10000002,15000' '10000003,15000' '10000004,15000' '10000005,15000' '10000006,15000' '10000007,15000' '10000008,15000' '10000009,15000' '10000010,15000' '10000011,15000' '10000012,15000' '10000013,15000' '10000014,15000' '10000015,15000' '10000016,15000' '10000017,15000' '10000018,15000' '10000019,15000' '10000020,15000' '10000021,15000' '10000022,15000' '10000023,15000' '10000024,15000' '10000025,15000' '10000026,15000' '10000027,15000' '10000028,15000' '10000029,15000' '10000030,15000' '10000031,15000' '10000032,15000' '10000033,15000' '10000034,15000' '10000035,15000' '10000036,15000' '10000037,15000' '10000038,15000' '10000039,15000' '10000040,15000' '10000041,15000' '10000042,15000' '10000043,15000' '10000044,15000' '10000045,15000' '10000046,15000' '10000047,15000' '10000048,15000' '10000049,15000' '10000050,15000' '10000051,15000' '10000052,15000' '10000053,15000' '10000054,15000' '10000055,15000' '10000056,15000' '10000057,15000' '10000058,15000' '10000059,15000' '10000060,15000' '10000061,15000' '10000062,15000' '10000063,15000' '10000064,15000' '10000065,15000' '10000066,15000' '10000067,15000' '10000068,15000' '10000069,15000' '10000070,15000' '10000071,15000' '10000072,15000' '10000073,15000' '10000074,15000' '10000075,15000' '10000076,15000' '10000077,15000' '10000078,15000' '10000079,15000' '10000080,15000' '10000081,15000' '10000082,15000' '10000083,15000' '10000084,15000' '10000085,15000' '10000086,15000' '10000087,15000' '10000088,15000' '10000089,15000' '10000090,15000' '10000091,15000' '10000092,15000' '10000093,15000' '10000094,15000' '10000095,15000' '10000096,15000' '10000097,15000' '10000098,15000' '10000099,15000')
for test in ${tests[@]}
do
    unset inputs
    IFS=',' read -r -a inputs <<< "$test"
    python3 ../../process_1.py ${inputs[0]} ${inputs[1]} | tee -a ../../Logs/Teste\ 4/output.txt
done
```

4.1. LOGS

Os logs da execução desses processos podem ser vistos no link abaixo:

<https://github.com/henriquevital00/multi-thread/tree/main/Logs/Teste%204>

5. REPOSITÓRIO DO GITHUB

Todo o código desenvolvido para o processo 1, 2 e 3, pode ser conferido no repositório abaixo, bem como os logs recebidos nos testes e os scripts de cada teste.

Link: <https://github.com/henriquevital00/multi-thread>