Task 1

```
→  Downloads ssh -i labsuser.pem ec2-user@3.90.242.182

A newer release of "Amazon Linux" is available.
  Version 2023.8.20250818:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Mon Sep  1 05:43:41 2025 from 124.43.209.182
[ec2-user@ip-172-31-33-192 ~]$ gcc -fopenmp ex1.c -o ex1
[ec2-user@ip-172-31-33-192 ~]$ ./ex1
Fibonacci(20) = 6765
Time: 0.008937 seconds
[ec2-user@ip-172-31-33-192 ~]$ OMP_NUM_THREADS=4 ./ex1
Fibonacci(20) = 6765
Time: 0.010870 seconds
[ec2-user@ip-172-31-33-192 ~]$  ./ex1.o > output_ex1.txt
-bash: ./ex1.o: No such file or directory
[ec2-user@ip-172-31-33-192 ~]$ ./ex1.o > output_ex1.txt
-bash: ./ex1.o: No such file or directory
[ec2-user@ip-172-31-33-192 ~]$ ./ex1.o > output_ex1.txt
-bash: ./ex1.o: No such file or directory
[ec2-user@ip-172-31-33-192 ~]$ ./ex1 > output_ex1.txt
[ec2-user@ip-172-31-33-192 ~]$
```

```
Last login: Mon Sep  1 11:10:34 on ttys001
→  ~ cd Downloads
→  Downloads nano ex1.c
→  Downloads scp -i ~/Downloads/labsuser.pem ~/Downloads/ex1.c ec2-user@3.90.242.182:/home/ec2-user/
ex1.c                                         100%  555     2.5KB/s   00:00
→  Downloads scp -i ~/Downloads/labsuser.pem ec2-user@3.90.242.182:/home/ec2-user/output_ex1.txt ~/Downloads/
→  Downloads scp -i ~/Downloads/labsuser.pem ec2-user@3.90.242.182:/home/ec2-user/output_ex1.txt ~/Downloads/
output_ex1.txt                                100%   44     0.1KB/s   00:00
→  Downloads
```

Task 2

```
Alternatively, you can use the --oversubscribe option to ignore the
number of available slots when deciding the number of processes to
launch.
--------------------------------------------------------------------------
[ec2-user@ip-172-31-33-192 ~]$ mpirun -np 1 ./sum_mpi
Total sum = 50000005000000
Time = 0.025384 seconds
[ec2-user@ip-172-31-33-192 ~]$ mpirun -np 1 ./sum_mpi > output_sum.txt
[ec2-user@ip-172-31-33-192 ~]$
```

```
→  Downloads nano sum_mpi.c
→  Downloads scp -i ~/Downloads/labsuser.pem ~/Downloads/sum_mpi.c ec2-user@3.90.242.182:/home/ec2-user/
sum_mpi.c                                     100%  813     3.1KB/s   00:00
→  Downloads scp -i ~/Downloads/labsuser.pem ec2-user@3.90.242.182:/home/ec2-user/output_sum.txt ~/Downloads/
output_sum.txt                                100%   51     0.1KB/s   00:00
→  Downloads
```

Task 3

```
[ec2-user@ip-172-31-33-192 ~]$ mpicc pi_mpi.c -o pi_mpi
[ec2-user@ip-172-31-33-192 ~]$ mpirun -np 1 ./pi_mpi
Approximate Pi = 3.141130
Time = 0.351209 seconds
[ec2-user@ip-172-31-33-192 ~]$ mpirun -np 1 ./pi_mpi > output_pi.txt
[ec2-user@ip-172-31-33-192 ~]$
```

```
→  Downloads nano pi_mpi.c
→  Downloads scp -i ~/Downloads/labsuser.pem ~/Downloads/pi_mpi.c ec2-user@3.90.
242.182:/home/ec2-user/
pi_mpi.c                                      100%  967     4.5KB/s   00:00
→  Downloads scp -i ~/Downloads/labsuser.pem ec2-user@3.90.242.182:/home/ec2-use
r/output_pi.txt ~/Downloads/
output_pi.txt                                 100%   50     0.1KB/s   00:00
→  Downloads
```

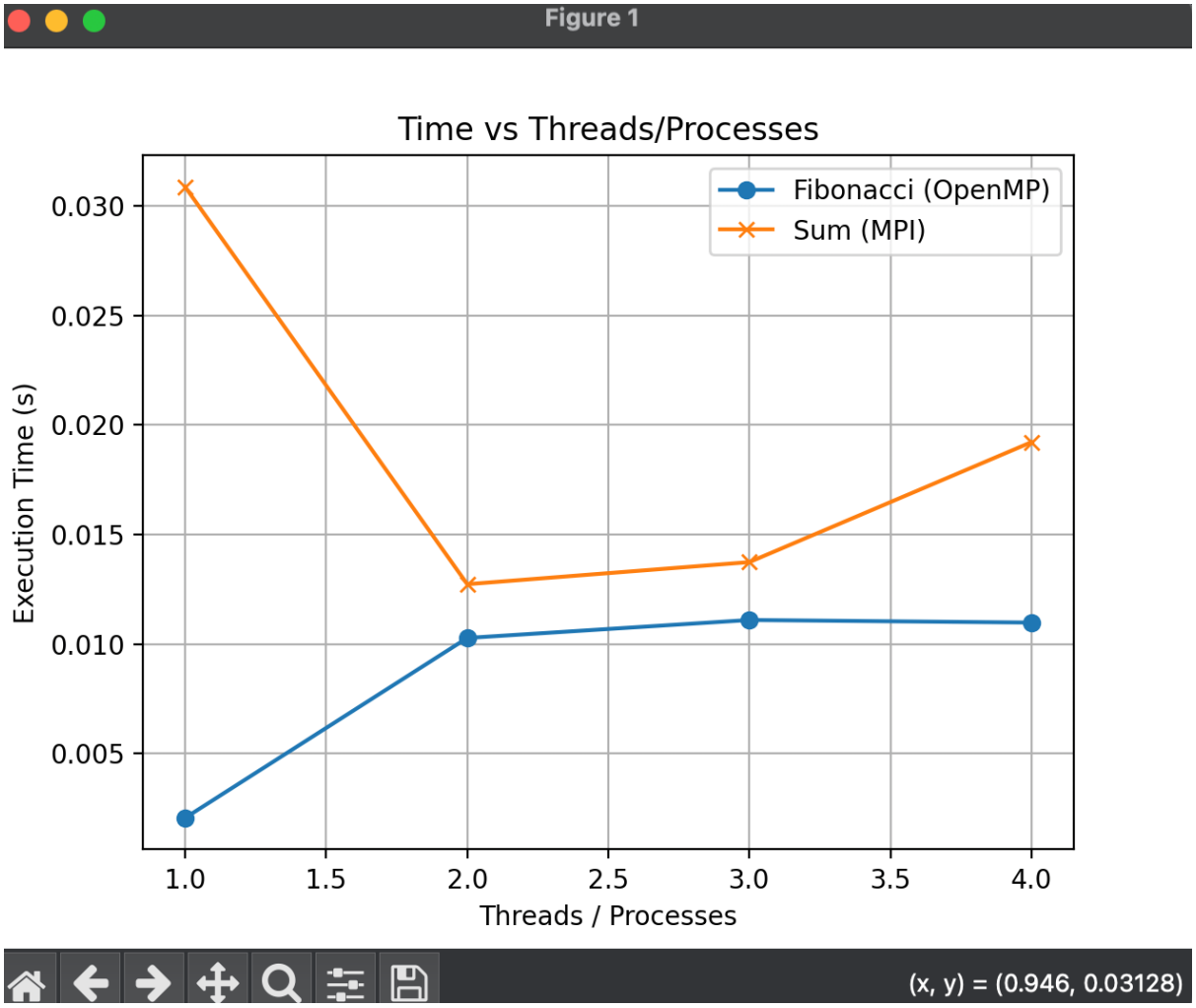
Task 4

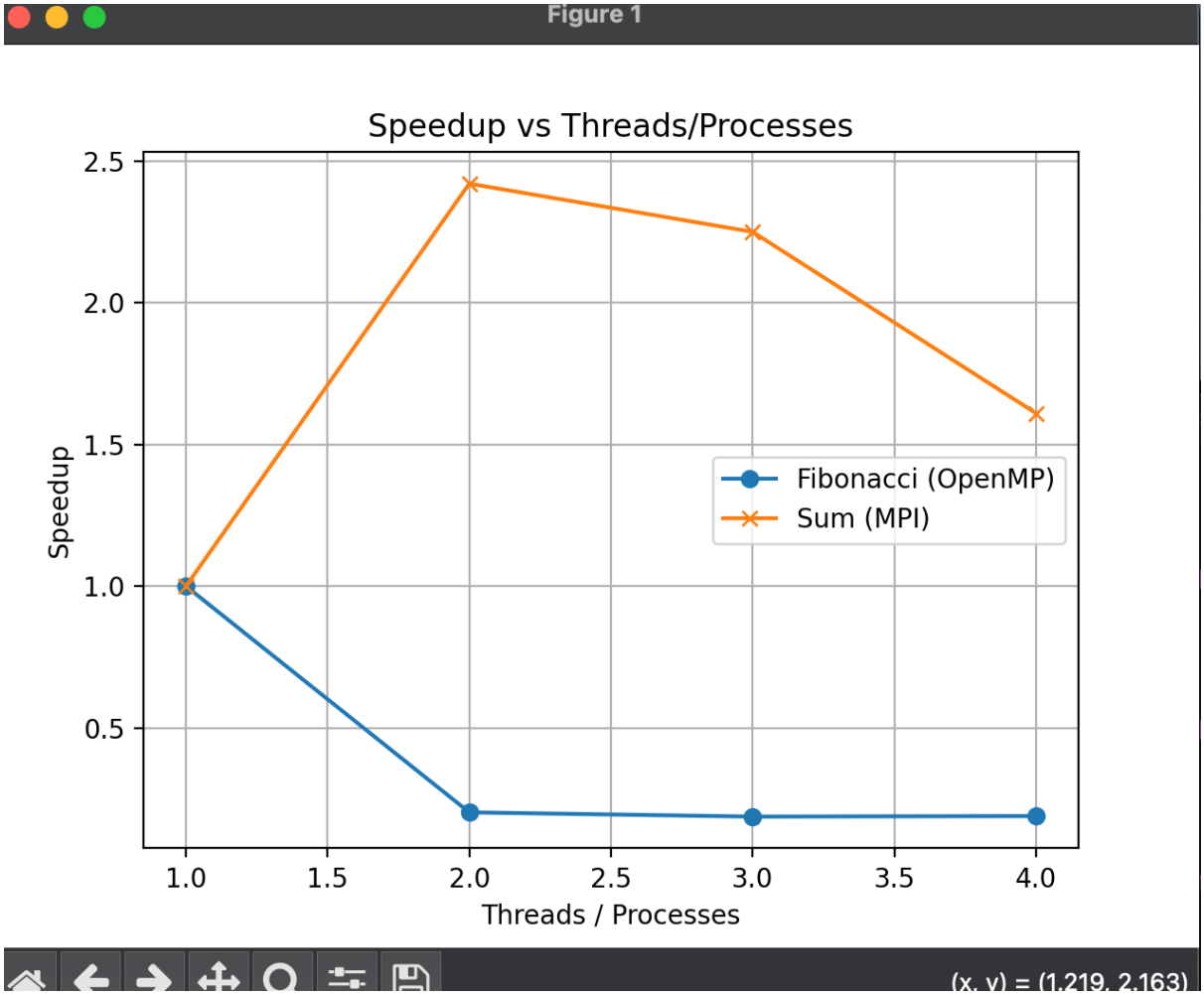
Fibonacci (OpenMP)

```
Last login: Mon Sep  1 11:24:48 2025 from 124.43.209.182
[ec2-user@ip-172-31-33-192 ~]$ gcc -fopenmp ex1.c -o ex1
[ec2-user@ip-172-31-33-192 ~]$ OMP_NUM_THREADS=1
[ec2-user@ip-172-31-33-192 ~]$ OMP_NUM_THREADS=1 ./ex1
Fibonacci(20) = 6765
Time: 0.002074 seconds
[ec2-user@ip-172-31-33-192 ~]$ OMP_NUM_THREADS=2 ./ex1
Fibonacci(20) = 6765
Time: 0.010282 seconds
[ec2-user@ip-172-31-33-192 ~]$ OMP_NUM_THREADS=3 ./ex1
Fibonacci(20) = 6765
Time: 0.011101 seconds
[ec2-user@ip-172-31-33-192 ~]$ OMP_NUM_THREADS=4 ./ex1
Fibonacci(20) = 6765
Time: 0.010984 seconds
[ec2-user@ip-172-31-33-192 ~]$
```

Sum (MPI)

```
[ec2-user@ip-172-31-33-192 ~]$ mpicc sum_mpi.c -o sum_mpi
[ec2-user@ip-172-31-33-192 ~]$ mpirun --oversubscribe -np 1 ./sum_mpi
Total sum = 50000005000000
Time = 0.030860 seconds
[ec2-user@ip-172-31-33-192 ~]$ mpirun --oversubscribe -np 2 ./sum_mpi
Total sum = 50000005000000
Time = 0.012735 seconds
[ec2-user@ip-172-31-33-192 ~]$ mpirun --oversubscribe -np 3 ./sum_mpi
Total sum = 50000005000000
Time = 0.013742 seconds
[ec2-user@ip-172-31-33-192 ~]$ mpirun --oversubscribe -np 4 ./sum_mpi
Total sum = 50000005000000
Time = 0.019216 seconds
[ec2-user@ip-172-31-33-192 ~]$
```





Task 6

```
Apple clang version 17.0.0 (clang-1700.0.13.5)
Target: arm64-apple-darwin24.6.0
Thread model: posix
InstalledDir: /Library/Developer/CommandLineTools/usr/bin
➜  ~ cd Downloads
➜  Downloads nano ex6_mismatch.c
➜  Downloads nano ex6_bsend.c
➜  Downloads nano ex7_anysource.c
➜  Downloads nano ex8_bsend.c
```

```
Last login: Tue Sep  2 21:29:58 on ttys002
➜  ~ cd downloads
➜  downloads mpicc ex6_mismatch.c -o ex6_mismatch
➜  downloads mpirun -np 3 ./ex6_mismatch

Process 0 sent message 42 to process 1
```

- When the **source and destination do not match** in MPI communication, the `MPI_Send` call from one process does not find a corresponding `MPI_Recv` in the expected target process. This causes the program to **hang indefinitely (deadlock)** because the message is sent but never received.
- In other words, if process 0 sends a message to process 1, but process 2 is waiting to receive from process 0, there is no matching communication pair. Since MPI requires matching **send–receive pairs**, the execution will not complete, and the program gets stuck.

```
Downloads — vidaththeekshana@Vidaths-MacBook-Air — ~/downloads — -zsh — 80×24
Last login: Tue Sep  2 21:40:58 on ttys003
➜  ~ cd downloads
➜  downloads mpicc ex6_bsend.c -o ex6_bsend
➜  downloads mpirun -np 2 ./ex6_bsend
Process 1 received message 100 from process 0
Process 0 buffered and sent message 100 to process 1
➜  downloads
```

Task 7

```
➜  downloads mpicc ex7_anysource.c -o ex7_anysource
➜  downloads mpirun -np 4 ./ex7_anysource

Process 1 sent message 10 to process 0
Process 2 sent message 20 to process 0
Process 0 received message 10
Process 0 received message 20
Process 3 sent message 30 to process 0
Process 0 received message 30
➜  downloads
```

Task 8

```
[➜  downloads mpicc ex8_bsend.c -o ex8_bsend
[➜  downloads mpirun -np 2 ./ex8_bsend
Process 1 received message 200 from process 0
Process 0 buffered and sent message 200 to process 1
➜  downloads
```

- Using MPI_ANY_SOURCE allows the receiver to accept messages from any process, so the program becomes more flexible. However, the order of received messages can change with each run, unlike the original fixed-source version which always received messages in the same order.