

# Programming Assignment 1

## I. Screenshots of outputs

The screenshot shows a VS Code editor with a C file named `time_shm.c`. The code includes standard headers and defines a shared memory location. The terminal output shows the execution of the program, comparing the time taken for message passing and shared memory communication. The output indicates that message passing is faster for the given task.

```
proj1 > C:\time_shm.c> share(long double)
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <fcntl.h>
5 #include <sys/shm.h>
6 #include <sys/stat.h>
7 #include <sys/mman.h>
8 #include <sys/time.h>
9 #include <sys/types.h>
10 #include <sys/wait.h>
11 #include <unistd.h>
12
13 //Resources:
14 //zybooks Section 3.7.1, Lecture 5 Process API Slides
15
16 //creating and writing to the shared memory location
17 void share(long double start){
18     const int size = 4096; //set size of shared memory
```

```
meolsen@hal:/mnt/c/Users/meols/OneDrive/Desktop/Mae/cs300/proj1$ ./time_shm ls -a
. . . a.out makefile submission submission1 time_pipe time_pipe.c time_shm time_shm.c
Elapsed time: 0.00418
meolsen@hal:/mnt/c/Users/meols/OneDrive/Desktop/Mae/cs300/proj1$ ./time_shm ls
a.out makefile submission submission1 time_pipe time_pipe.c time_shm time_shm.c
Elapsed time: 0.00732
meolsen@hal:/mnt/c/Users/meols/OneDrive/Desktop/Mae/cs300/proj1$ ./time_shm mkdir submission
Elapsed time: 0.00576
meolsen@hal:/mnt/c/Users/meols/OneDrive/Desktop/Mae/cs300/proj1$ ./time_pipe ls -a
. . . a.out makefile submission time_pipe time_pipe.c time_shm time_shm.c
Elapsed time: 0.00887
meolsen@hal:/mnt/c/Users/meols/OneDrive/Desktop/Mae/cs300/proj1$ ./time_pipe ls
a.out makefile submission time_pipe time_pipe.c time_shm time_shm.c
Elapsed time: 0.00688
meolsen@hal:/mnt/c/Users/meols/OneDrive/Desktop/Mae/cs300/proj1$ ./time_pipe mkdir submission1
Elapsed time: 0.00597
meolsen@hal:/mnt/c/Users/meols/OneDrive/Desktop/Mae/cs300/proj1$
```

II. Based on my observations, message passing is faster than shared memory for each command I've tried because the time being shared is a relatively small amount of information to communicate.

### A. Pros/Cons for shared memory

Pros:	Cons:
<ul style="list-style-type: none"><li>Can be faster than message passing because no kernel intervention after object creation</li></ul>	<ul style="list-style-type: none"><li>Controlling synchronization and security issues</li></ul>

### B. Pros/Cons for message passing

Pros:	Cons:
<ul style="list-style-type: none"><li>Good for exchanging smaller amounts of data because collisions aren't an issue</li></ul>	<ul style="list-style-type: none"><li>Implemented using system calls, so kernel intervention requires more time than shared memory</li></ul>
<ul style="list-style-type: none"><li>Easier to implement than shared memory</li></ul>	
<ul style="list-style-type: none"><li>Easier to use in distributed systems</li></ul>	