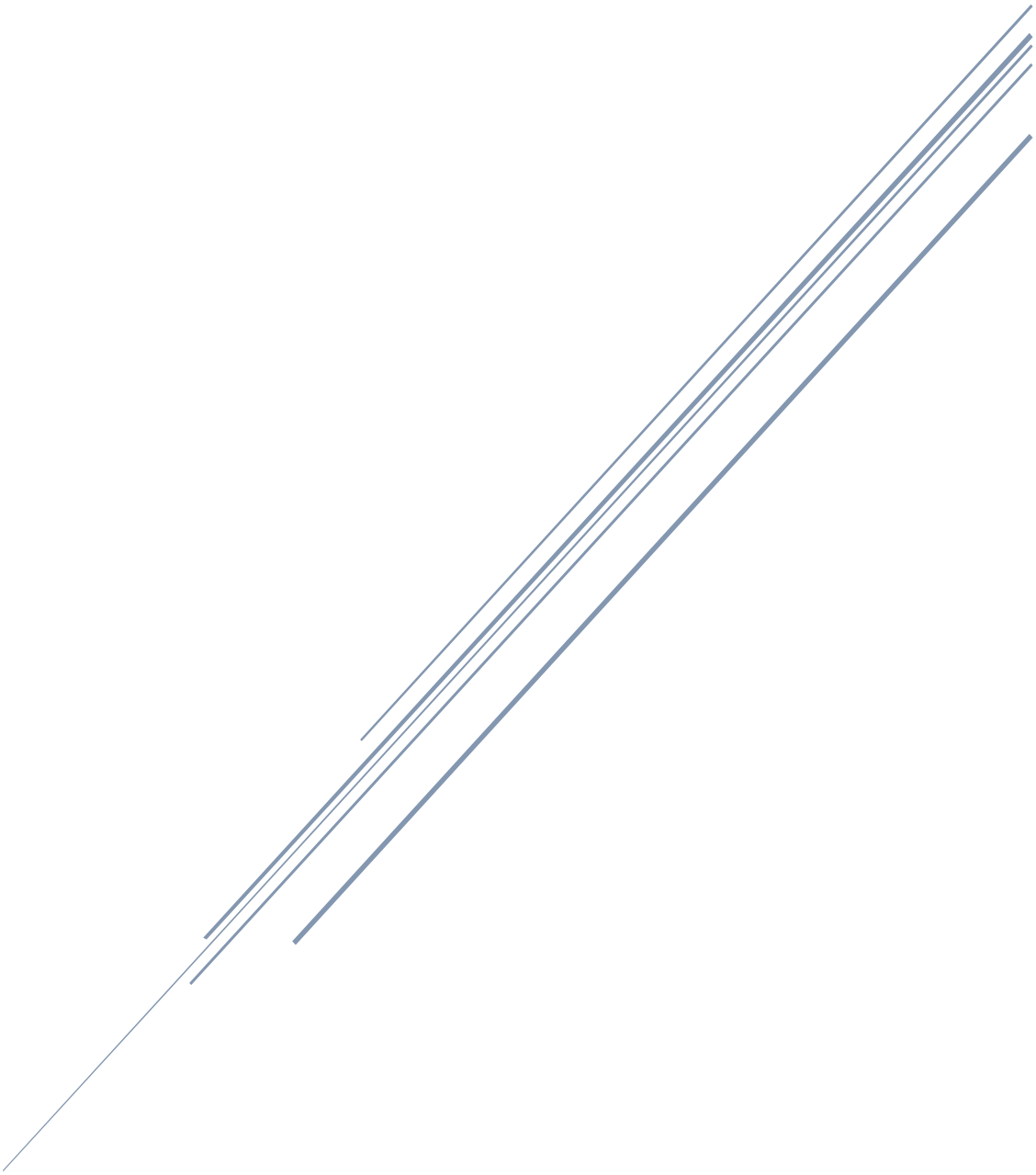
PROJECT 1 - REPORT

OPERATING SYSTEMS



MEHUL SURESH KUMAR
U52982215

Program 1

Process.c – measures the time it takes to create a process.

AIM: To repeatedly call a fork() command to create a new process and measure the time it takes to create a new process

How to Run:

- make
- ./process
- Enter the number of iteration to run the test

Sample execution:

[mehul1@c4lab12 os]$ ./process

Enter the number of iterations to run the test :7

Creating process 31311 which completes after 80 microseconds

Creating process 31312 which completes after 105 microseconds

Creating process 31313 which completes after 126 microseconds

Creating process 31314 which completes after 125 microseconds

Creating process 31315 which completes after 159 microseconds

Creating process 31316 which completes after 190 microseconds

Creating process 31317 which completes after 243 microseconds

---

Average time to create a process is 34 microseconds

Description:

- The program first initializes the data structures required to utilizes the gettimeofday() function
- After which gettimeofday() is used to get a time stamp to be used as a reference value
- The fork is created as many times as the user requests it
- The creation time of each new process is stored
- After creating the required number of processes the total time needed to create the processes is divided by the number of processes to derive the average process creation time

Thread.c

Aim: To repeatedly create a thread which executes concurrently and measure the time it takes to create a new thread.

How to Run:

- make
- ./thread
- Enter the number of threads to be created

Sample Execution:

Enter the number of iterations to run the test :10

Thread 1 took 90 microseconds to be created

Thread 2 took 40 microseconds to be created

Thread 3 took 41 microseconds to be created

Thread 4 took 41 microseconds to be created

Thread 5 took 32 microseconds to be created

Thread 6 took 17 microseconds to be created

Thread 7 took 17 microseconds to be created

Thread 8 took 16 microseconds to be created

Thread 9 took 11 microseconds to be created

Thread 10 took 11 microseconds to be created

Average time taken for thread creation is is 31.60 microseconds

Note:

The threads and processes created in the beginning generally take much longer than the ones created later during the execution of the program.

Description:

- gettimeofday() function is initialized to get the time stamp.

- I create the thread reference variable and the entry point function.
- The thread is created and joined
- The time it enters the function is recorded and compared with the reference time stamp.
- The average is calculated at the very end and displayed.

Context_switch.c

Aim:

Create two processes and two pipes so that they can communicate. Make them mutually wait for each other to proceed and measure the average time it takes to switch between them.

How to Run:

- make
- ./context_switch
- Enter the number of iterations

Sample Execution:

[mehul1@c4lab12 os]$ ./context_switch

Enter the number of iterations to run the test

5

Two pipes have been created

Process 1 ran for 28 ms

Message Received by Process 2: Process 1 says 'Hi!'

Process 2 ran for 25 ms

Message Received by Process 1: Process 2 says 'Hello!'

----------ONE CYCLE COMPLETE----------

Process 1 ran for 19 ms

Message Received by Process 2: Process 1 says 'Hi!'

Process 2 ran for 7 ms

Message Received by Process 1: Process 2 says 'Hello!'

----------ONE CYCLE COMPLETE----------

Process 1 ran for 7 ms

Message Received by Process 2: Process 1 says 'Hi!'

Process 2 ran for 6 ms

Message Received by Process 1: Process 2 says 'Hello!'

----------ONE CYCLE COMPLETE----------

Process 1 ran for 7 ms

Message Received by Process 2: Process 1 says 'Hi!'

Process 2 ran for 6 ms

Message Received by Process 1: Process 2 says 'Hello!'

----------ONE CYCLE COMPLETE----------

Process 1 ran for 7 ms

Message Received by Process 2: Process 1 says 'Hi!'

Average Context Switching time of Process 2 is :13.60<----------

FINISHED PROCESS 2

Process 2 ran for 62 ms

Message Received by Process 1: Process 2 says 'Hello!'

----------ONE CYCLE COMPLETE----------

Average Context Switching time of Process 1 is :21.20<----------

FINISHED PROCESS 1

Notes: The first and the last iterations of the process switching make much longer than the others possibly due to the overhead of opening and closing the pipes.

Description:

- First I make sure that the program executes on a single processor as running it on multiple processors may corrupt the results
- Then two pipes are created to handle communications between the two processes
- Two child processes are created
- The user is asked the number of iterations.
- Each process closes the pipes it won't need to use
- The first process writes a message to pipe 1 and then waits to read on pipe 2
- The second process waits to read from pipe 1 and writes a message to pipe2
- The cycle repeats
- The time it's to wait is recorded
- The final average time each process waits is displayed before it quits.
- The used pipes are closed.