Learning Report

Linux OS & Programming

LTTS GLOBAL ENGINEERING ACADEMY

L&T Technology Services

Document History

Ver. Rel. No.	Release Date	Prepared. By	Reviewed By	To be Approved By	Remarks/Revision Details
1	1-03-21	N Prakruthi			
2	2-03-21	N Prakruthi			
3	4-03-21	N Prakruthi			

Activities

Activity 1 – Introduction to Linux OS and Libraries linking

Learning Resources:

https://embetronicx.com/tutorials/unit_testing/unit-testing-in-c-testing-with-unity
https://www3.ntu.edu.sg/home/ehchua/programming/cpp/gcc_make.html

Goal of Activity: Static library and Dynamic library. Creating user defined libraries and linking user defined functions as library both statically and dynamically.

Topics covered: Linux OS Architecture, GCC & Build Process, Utilities, Static & Dynamic Libraries, Makefile creation, GCC & Build Process.

Learning Outcomes:Activity based on creating multiple functions with many operations like string compare, concatenation and creating makefiles depending on the conditions.

Challenges: Implementing and remembering of System calls and Processes related commands

Github link: https://github.com/prakruthin/linux_3551

Activity 2:
Type of Activity: Individual

Goal of Activity: Linux OS architecture

Topics covered: Linux OS Architecture, Utilities, Static and dynamic libraries, Makefile creation & Build process Learning

Outcomes: Implemented the working of Stages in scheduling of processes, Zombie processes system calls and signals, Context switch and structure of Linux OS

Challenges : Implementing System calls and Processes related commands

Learning Resources:
https://web.microsoftstream.com/video/9e33e60e-91e3-4b6f-ac23-937e83897e86
https://www3.ntu.edu.sg/home/ehchua/programming/cpp/gcc_make.html

Activity 3: Semaphores and Mutex

Type of Activity: Individual

Goal of Activity: Implement producer consumer problem

Topics Covered:

Mutex Lock, Semaphores- Named and unnamed, Race condition, Deadlock, Pipes, Shared memory, Message queue

Learning Outcomes:

- Working with named and unnamed semaphores, and using named semaphores in shared memory.
- Handling context switching in order to avoid deadlocks.
- Analyzing the return type for mutex to check for success or failure.
- Using operations on shared memory such as read write and update.
- Learnt to implement sequencing and mutual exclusion.

Challenges: Understanding the deadlocks and shared memory

References:

[1] https://www.tutorialspoint.com/gnu_debugger/index.htm
[2] https://www3.ntu.edu.sg/home/ehchua/programming/cpp/gcc_make.html
[3] https://tutorialspoint.com/operating_system/os_linux.htm