./

Learning Report-Linux OS and Programming



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **Approved By** | **Remarks/Revision Details** |
| 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Document History**

Contents

[Contents 3](#_Toc65876498)

[Activity 1: Perform functions along with static and dynamic linking (01.03.2021) 4](#_Toc65876499)

[Activity 2: Perform operations using System calls, Signals and Processes (02.03.2021) 5](#_Toc65876500)

[Activity 3: Perform operations using Shell commands and Threads (03.03.2021) 6](#_Toc65876501)

[Activity 4: Inter Process Communication related problems (04.03.2021) 7](#_Toc65876502)

[Activity 5: Perform operations implementing Message Queues and Pipes (05.03.2021) 8](#_Toc65876503)

# Activity 1: Perform functions along with static and dynamic linking (01.03.2021)

**Type of Activity**: Individual

**Goal of Activity**: Perform different functions for different data types and adding static and dynamic libraries.

**Topics covered:** Architecture of OS, Types of OS, GCC & Build Process, Static and Dynamic Libraries, Makefile creation.

**Learning Outcomes:** Able to write code without using scanf and printf statements, adding makefiles, static and dynamic linking of libraries.

**Challenges:** Difficulty in implementing the static and dynamic linking.

**GitHub Link:**

* <https://github.com/GraceJoseph07/Linux_Activities/tree/main/Day1_01.03.2021/Individual_Activity>

**Learning Resources:**

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

**References:**

* https://web.microsoftstream.com/video/9a2b1eba-61a3-4547-8292-374b2eeb5265?channelId=04fdad23-021c-4e64-bb7c-06b2469801f9
* <https://web.microsoftstream.com/video/5cc492de-e71c-4c15-98ff-53727580a5b6?channelId=04fdad23-021c-4e64-bb7c-06b2469801f9>

# 

# Activity 2: Perform operations using System calls, Signals and Processes (02.03.2021)

**Type of Activity**: Individual

**Goal of Activity**: To understand and implement programs using System calls, Signals and Processes.

**Topics covered:** Linux OS Architecture, Interrupts, System calls, Scheduling, Signals, Process Life Cycle, Context Switch, Process related commands.

**Learning Outcomes:** Understanding the working of system calls and signals, structure of Linux OS, stages in scheduling of processes, Context saving and loading and process related commands.

**Challenges:** Remembering all the commands related to System calls and Processes.

**GitHub Link:**

* <https://github.com/GraceJoseph07/Linux_Activities/tree/main/Day2_02.03.2021>

**Learning Resources:**

* https://www.tutorialspoint.com/operating\_system/os\_quick\_guide.htm
* <https://www.geeksforgeeks.org/input-output-system-calls-c-create-open-close-read-write/>
* <https://www.cs.uregina.ca/Links/class-info/330/SystemCall_IO/SystemCall_IO.html#FileIO>

**References:**

* <https://www.csl.mtu.edu/cs4411.ck/www/NOTES/process/fork/create.html>
* <https://linuxhint.com/linux-exec-system-call/>

# Activity 3: Perform operations using Shell commands and Threads (03.03.2021)

**Type of Activity**: Individual

**Goal of Activity**: To understand the concept of threads and shell commands.

**Topics covered:** Zombie, Orphan and Daemon Process, Threads and Unit testing

**Learning Outcomes:** Understand thread concepts and it’s types, working of CPU during Context Switching and the concepts of Zombie process, Orphan process and Demon process. Implementation of test files

**Challenges:** Keeping track of different shell commands with syntax.

**GitHub Link:**

* <https://github.com/GraceJoseph07/Linux_Activities/tree/main/Day2_02.03.2021>

**Learning Resources:**

* <https://www.geeksforgeeks.org/zombie-and-orphan-processes-in-c/>
* <https://www.tutorialspoint.com/zombie-vs-orphan-vs-daemon-processes>
* <https://www.geeksforgeeks.org/basic-shell-commands-in-linux/>

**References:**

* <https://www.thegeekstuff.com/2012/03/linux-threads-intro/>
* <https://www.cs.cmu.edu/afs/cs/academic/class/15492-f07/www/pthreads.html>

# Activity 4: Inter Process Communication related problems (04.03.2021)

**Type of Activity**: Individual

**Goal of Activity**: To understand the concepts of Inter Process Communication.

**Topics covered:** Semaphores and Mutex.

**Learning Outcomes:** Understand the concepts to prevent race around conditions and scheduling issues using Semaphores and Mutual Exclusion in critical sections.

**Challenges:** Difficulty in following through the coding parts in IPC

**GitHub Link:**

* <https://github.com/GraceJoseph07/Linux_Activities>

**Learning Resources:**

* <https://www.geeksforgeeks.org/inter-process-communication-ipc/>
* <https://www.guru99.com/inter-process-communication-ipc.html>
* <https://www.guru99.com/semaphore-in-operating-system.html>

**References:**

* <https://opensource.com/article/19/4/interprocess-communication-linux-storage>
* <https://www.tutorialspoint.com/mutex-vs-semaphore>

# Activity 5: Perform operations implementing Message Queues and Pipes (05.03.2021)

**Type of Activity**: Individual

**Goal of Activity**: To understand the concept of pipes and message queues.

**Topics covered:** Pipes and it’s limitations, Shared memory and Message Queues.

**Learning Outcomes:** Understand the concepts of pipes and working of inline inputs, working of shared memory and it’s types along with commands and Message Queues.

**Challenges:** Keeping track of different commands with syntax.

**GitHub Link:**

* <https://github.com/GraceJoseph07/Linux_Activities>

**Learning Resources:**

* <https://www.geeksforgeeks.org/pipe-system-call/>
* <https://www.tutorialspoint.com/inter_process_communication/inter_process_communication_pipes.htm>
* <https://www.geeksforgeeks.org/ipc-using-message-queues/>

**References:**

* <https://www.tutorialspoint.com/inter_process_communication/inter_process_communication_shared_memory.htm>
* <https://www.tutorialspoint.com/inter_process_communication/inter_process_communication_message_queues.htm>