

CS-2006 Operating Systems Project List (Spring-2024)

The following list is project ideas that can be selected.

S. no.	Project name
1	System Call for Semaphore (Example: Reader-Writer Problem)
2	System Call for Semaphore- Chain Smoker Problem
3	Ice Cream Factory Problem (System call)
4	Dining Philosophers Problem (System Call)
5	Spin Lock (System Call)
6	Sleeping Barber problem (system call)
7	Real-Time Scheduling on Raspberry Pi/ MCU32
8	Cache Cleaner (No scripting)
9	Voice-controlled Shell (Without python library)
10	Modification of System Call - Open () and Close() / Must display additional information
11	Preemptive Scheduler on kernel Threads
12	Implementation of A* Search Algorithm with Threads
13	Comparison between Process and Threads (Ex. merge, quicksort with process and threads, any five algorithms)
14	Process Communication between sockets with GUI
15	Parallel Programming Comparison of sorting Algorithms using Pthreads vs. OpenMP vs. serial, (3 algorithms)
16	Development of Unix Shell (Combining commands)
17	Lightweight Virtual Machine
18	Task Manager implementing at least 5 customized operations
19	Page replacement algorithm by kernel threads

20	Comparison between Process and Threads in Android (Ex. merge, quicksort with process and threads, any five algorithms)
21	Comparisons of IPC mechanisms on Android
22	Shell script for any sensor using RaspberryPi/MCU32
23	Parallel Programming in Android/iOS Mobile app - Comparison of sorting algorithms using threads (3 Sorting algorithms)
24.	Hardware-based synchronization implementation using Semaphore
25.	Application of embedded Linux using QEMU
26.	Application based on any open-source RTOS
27.	Application of any Linux-based system-level multicore programming.

Note: For system call projects, kernel configuration should be done with student ID.

Requirements:

1. Group Instructions:

- a. Maximum 3 Group members
- b. Cross-section of the same teacher of the class is allowed ONLY
- c. The same project is not allowed within a section and approval will be granted on an FCFS basis.
- d. Please note that there is only one project in the course. project demo and viva marks will be put in theory.
- e. Only C or C ++ language preferred. **No python-based projects in OS.**

2. Deliverables and Due Dates:

- a. Project Proposal (project title, introduction, methodology) submission on **18th March 2024**
- b. Final Demonstration with the following requirements:
 - i. Final Working Project Demo
 - ii. Project Viva
 - iii. Final Report soft copy (objectives, project details, results (comparison via graphs), conclusion.
 - iv. Code repository on GitHub

Note: Final project demonstration is tentatively scheduled from 29th April 2024.