# SAVEETHASCHOOLOFENGINEERING

**SAVEETHAINSTITUTE OFMEDICAL AND TECHNICAL SCIENCESCOMPUTERSCIENCEANDENGINEERINGPROGRAMME**

# CSA04–OPERATINGSYSTEMS

**LISTOFPROGRAMS (DAY 1)**

1. Create a new process by invoking the appropriate system call. Get the process identifier of thecurrently running process and its respective parent using system calls and display the sameusinga Cprogram.
2. Identify the system calls to copy the content of one file to another and illustrate the same usinga Cprogram.
3. Design a CPU scheduling program with C using First Come First Served technique with thefollowingconsiderations.
   1. All processes are activated at time0.
   2. AssumethatnoprocesswaitsonI/Odevices.
4. ConstructaschedulingprogramwithCthatselectsthewaitingprocesswiththesmallestexecutiontimetoexecute next.
5. ConstructaschedulingprogramwithCthatselectsthewaitingprocesswiththehighestprioritytoexecutenext.
6. ConstructaCprogramtoimplementpre-emptivepriorityschedulingalgorithm.
7. ConstructaCprogramtoimplementnon-preemptiveSJFalgorithm.
8. ConstructaCprogramtosimulateRoundRobinschedulingalgorithmwithC.

# SAVEETHASCHOOLOFENGINEERING

**SAVEETHAINSTITUTE OFMEDICAL AND TECHNICAL SCIENCESCOMPUTERSCIENCEANDENGINEERINGPROGRAMME**

# CSA04–OPERATINGSYSTEMS

**LISTOFPROGRAMS (DAY 2)**

1. Illustrate the concept of inter process communication using message queue with a C program.
2. Illustrate the concept of inter-process communication using shared memory with a

C program.

1. Illustrate the concept of multithreading using a C program.
2. Design a C program to simulate the concept of Dining-Philosophers problem
3. Construct a C program for implementation the various memory allocation strategies.
4. Construct a C program to organize the file using single level directory.
5. Design a C program to organize the file using two level directory structures.
6. Develop a C program for implementing and random access file for processing the employee details

# SAVEETHASCHOOLOFENGINEERING

**SAVEETHAINSTITUTE OFMEDICAL AND TECHNICAL SCIENCESCOMPUTERSCIENCEANDENGINEERINGPROGRAMME**

# CSA04–OPERATINGSYSTEMS

**LISTOFPROGRAMS (DAY 3)**

1. Illustrate the deadlock avoidance concept by simulating Banker’s algorithm with C.
2. Construct a C program to simulate producer-consumer problem using semaphores.
3. Design a C program to implement process synchronization using mutex locks.
4. Construct a C program to simulate Reader-Writer problem using Semaphores.
5. Develop a C program to implement worst fit algorithm of memory management.
6. Construct a C program to implement best fit algorithm of memory management.
7. Construct a C program to implement first fit algorithm of memory management.
8. Design a C program to demonstrate UNIX system calls for file management.

# SAVEETHASCHOOLOFENGINEERING

**SAVEETHAINSTITUTE OFMEDICAL AND TECHNICAL SCIENCESCOMPUTERSCIENCEANDENGINEERINGPROGRAMME**

# CSA04–OPERATINGSYSTEMS

**LISTOFPROGRAMS (DAY 4)**

1. ConstructaCprogramtoimplementtheI/OsystemcallsofUNIX(fcntl,seek,stat,opendir,readdir)
2. ConstructaCprogramtoimplementthefilemanagementoperations.
3. DevelopaCprogramforsimulatingthefunctionoflsUNIXCommand.
4. WriteaCprogramforsimulationofGREPUNIXcommand
5. WriteaCprogramtosimulatethesolutionofClassicalProcessSynchronizationProblem
6. WriteCprogramstodemonstratethefollowingthreadrelatedconcepts.

(i)create(ii)join(iii)equal(iv)exit

1. ConstructaCprogramtosimulatetheFirstinFirstOutpagingtechniqueofmemorymanagement.
2. Construct a C program to simulate the Least Recently Used paging technique of memorymanagement.

# SAVEETHASCHOOLOFENGINEERING

**SAVEETHAINSTITUTE OFMEDICAL AND TECHNICAL SCIENCESCOMPUTERSCIENCEANDENGINEERINGPROGRAMME**

# CSA04–OPERATINGSYSTEMS

**LISTOFPROGRAMS (DAY 5)**

1. ConstructaCprogramtosimulatetheoptimalpagingtechniqueofmemorymanagement
2. Considerafilesystemwheretherecordsofthefilearestoredone afteranotherbothphysicallyandlogically.Arecordofthefilecanonlybeaccessedbyreadingallthepreviousrecords.DesignaCprogramtosimulatethefileallocationstrategy.
3. Consider a file system that brings all the file pointers together into an index block. The ithentry in the index block points to the ith block of the file. Design a C program to simulate the fileallocationstrategy.
4. Withlinkedallocation,each fileisalinkedlistofdisk blocks;thediskblocks maybescatteredanywhere on the disk. The directory contains a pointer to the first and last blocks of the file.Eachblock contains a pointer to the next block. Design a C program to simulate the file allocationstrategy.
5. ConstructaCprogramtosimulatetheFirstComeFirstServeddiskschedulingalgorithm.
6. DesignaCprogramtosimulateSCANdiskschedulingalgorithm.
7. DevelopaCprogramtosimulateC-SCANdiskschedulingalgorithm.
8. IllustratethevariousFileAccessPermissionanddifferenttypesusersinLinux.