Operating system (BHCS06) Discipline Specific Core Course - (DSC)

Credit: 06

Course Objective

The course introduces the students to different types of operating systems. Operating system modules such as memory management, process management and file management are covered in detail.

Course Learning Outcomes

On successful completion of the course, the students will be able to:

1. Implement multiprogramming, multithreading concepts for a small operating system.
2. Create, delete, and synchronize processes for a small operating system.
3. Implement simple memory management techniques.
4. Implement CPU and disk scheduling algorithms.
5. Use services of modern operating system efficiently
6. Implement a basic file system.

Detailed Syllabus

Unit 1

Introduction: Operating systems (OS) definition, Multiprogramming and Time Sharing operating systems, real time OS, Multiprocessor operating systems, Multicore operating systems, Various computing environments.

Unit 2

Operating System Structures: Operating Systems services, System calls and System programs, operating system architecture (Micro Kernel, client server) operating

Unit 3

Process Management: Process concept, Operation on processes, Multi-threaded processes and models, Multicore systems, Process scheduling algorithms, Process synchronization. The Critical-section problem and deadlock characterization, deadlock handling.

Unit 4

Memory Management: Physical and Logical address space; Memory allocation strategies - Fixed and Variable Partitions, Paging, Segmentation, Demand Paging and virtual memory, Page Replacement algorithm.

Unit 5

File and I/O Management: Directory structure, File access methods, Disk scheduling algorithms.

Practical

1. Write a program (using fork() and/or exec() commands) where parent and child execute: a) same program, same code. b) same program, different code. - c) before terminating, the parent waits for the child to finish its task.
2. Write a program to report behaviour of Linux kernel including kernel version, CPU type and model. (CPU information)
3. Write a program to report behaviour of Linux kernel including information on 19 configured memory, amount of free and used memory. (memory information)
4. Write a program to print file details including owner access permissions, file access time, where file name is given as argument.
5. Write a program to copy files using system calls.
6. Write a program to implement FCFS scheduling algorithm.
7. Write a program to implement Round Robin scheduling algorithm.
8. Write a program to implement SJF scheduling algorithm.
9. Write a program to implement non-preemptive priority based scheduling algorithm.
10. Write a program to implement preemptive priority based scheduling algorithm.
11. Write a program to implement SRJF scheduling algorithm.
12. Write a program to calculate sum of n numbers using thread library.
13. Write a program to implement first-fit, best-fit and worst-fit allocation strategies.

References

1. Silberschatz, A., Galvin, P. B., & Gagne, G. (2008). *Operating Systems Concepts*. 8th edition.. John Wiley Publications.

Additional Resources

1. Dhamdhere, D. M. (2006). *Operating Systems: A Concept-based Approach*. 2nd edition. Tata McGraw-Hill Education.

2. Kernighan, B. W., & Rob Pike, R. (1984). *The Unix programming environment* (Vol. 270). Englewood Cliffs, NJ: Prentice-Hall

3. Stallings, W. (2018). *Operating Systems: Internals and Design Principles*. 9th edition. Pearson Education.

4. Tanenbaum, A. S. (2007). *Modern Operating Systems*. 3rd edition. Pearson Education.

Course Teaching Learning Process

- Use of ICT tools in conjunction with traditional class room teaching methods
- Interactive sessions
- Class discussions

Tentative weekly teaching plan is as follows:

Week	Content
1	Operating System, Definition and its purpose, Time sharing, Multiprogramming and Multiprocessing, Operating System Operations
2	Operating System Services, User and Operating System Interface, System Calls and its Types.
3	Operating system Design and Structure, System Programs, System Boot, Process
4	Operations on Processes, Inter process communication, Shared memory.
5	Multithreading Models, Multicore Programming, Thread Libraries
6	Process Scheduling criteria, Process Scheduling Algorithms, Multiple Processor Scheduling.
7	Process Synchronization, Critical Section Problem, Semaphores.
8	Deadlock Characterization, Methods for handling deadlocks.
9-10	Memory Allocation Strategies-Fixed and Variable partition, Swapping, Logical and Physical Address Space, Paging, Structure of Page Table and its Variations, Shared pages, Segmentation
11-12	Virtual memory, Page Replacement Algorithms, Allocation of frames, Thrashing, Working set model.

13-14	File System , File Characteristics, Access methods, Directory and Disk structure , File system structure and implementation, Directory implementation, Free space Implementation, File Allocation methods.
15	Overview of Secondary Devices, Disk Scheduling Algorithms

Assessment Methods

Written tests, assignments, quizzes, presentations as announced by the instructor in the class.

Keywords

Types of operating systems, memory management, process management, file and I/O management

---

Computer Networks (BHCS07) Discipline Specific Core Course - (DSC)

Credit: 06

Course Objective

This course covers the concepts of data communication and computer networks. It comprises of the study of the standard models for the layered protocol architecture to communicate between autonomous computers in a network and also the main features and issues of communication protocols for different layers. Topics covered comprise of introduction to OSI and TCP/IP models also.

Course Learning Outcomes

On successful completion of the course, the student will be able to:

1. Describe the hardware, software components of a network and their interrelations.
2. Compare OSI and TCP/IP network models.
3. Describe, analyze and compare different data link, network, and transport layer protocols.
4. Design/implement data link and network layer protocols in a simulated networking environment.