



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI WORK INTEGRATED LEARNING PROGRAMMES

COURSE HANDOUT

Part A: Content Design

Course Title	Systems Programming
Course No(s)	CSI* ZC327 / SE* ZC333 / SE* ZC332 / IS ZC327
Credit Units	3 / 4
Credit Model	3 (1-1-1) / 4 (1-1-2) (1 unit for class room hours, 1 unit for lab hours, 1 unit for student preparation (2 units if 4 credit course)). Typically 1 unit translates to 32 hours
Course Author	Chandra Shekar RK
Version No	1.0
Date	

Course Objectives

No	Course Objective
CO1	To provide understanding of the files, directory and file systems.
CO2	To provide insight into UNIX commands and Shell scripting.
CO3	To provide understanding of the System calls and its usage with C programming.
CO4	To provide understanding of assemblers, linkers and loaders

Text Book(s)

T1	Brian. W. Kernighan and Rob Pike, “The UNIX Programming Environment”, Prentice – Hall of India, Pvt. Ltd, 1984.
----	---

Reference Book(s) & other resources

R1	Muster J.C., Introduction to UNIX and LINUX, McGraw Hill, 2002.
R2	Quigley E., UNIX Shells by Example, Prentice Hall, 2001.
R3	O'Reilly, Learning the vi Editor, 6th Edition, 1998, Arnold Robbins, Linda Lamb

R4	O'Reilly, Learning the bash Shell, 2nd Edition, By Cameron Newham, Publisher: O'Reilly Media, Released: March 2005, 3rd edition
R5	W. Richard Stevens, "Advanced Programming in the UNIX environment" 2nd edition, Addition-Wesley, 2005.
R6	Maurice J. Bach, "The Design of the Unix Operating System", PHI, 1993
R7	Leland L. Beck, "System Software – An Introduction to Systems Programming", 3rd Edition, Pearson Education Asia, 2006.

Modular Content Structure

1. **Directories and Files in UNIX / Linux system. Basic UNIX commands**
 - 1.1. Getting Introduction to UNIX / Linux
 - 1.2. Logging in to Linux system
 - 1.3. UNIX files and directories
 - 1.4. Basic UNIX commands
2. **Internals of Unix file system, I/O in UNIX**
 - 2.1. Basics of files, UNIX file system, Inode structure in UFS, Links in UNIX
 - 2.2. Directories in UNIX, Conversion of a path name to an inode, open files and descriptor management
 - 2.3. File system layout, Superblock, I/O in UNIX – Kernel I/O structure, I/O devices – Block and Character devices, Device drivers, I/O queuing and interrupt handling
3. **VI editor, File and directory commands, I/O indirection and Filters**
 - 3.1. Working with VI editor: Basic operating modes
 - 3.2. Commands for copying & pasting text and pattern searching & substitution of text in command mode, Recovering files in vi editor
 - 3.3. File and directory commands – cat, cp, mv, rm, chmod, cmp, diff and top
 - 3.4. I/O indirection, Filters – wc, sort, head, tail, grep, pipe and tee
4. **Shell Scripting**
 - 4.1. Pattern matching and Shell Meta characters
 - 4.2. Shell Scripting: simple programs, interactive shell scripting, arithmetic in shell, variables in shell with more example programs
 - 4.3. Decision making constructs in Shell with more example programs
 - 4.4. Loop constructs in Shell with more example programs
 - 4.5. Files using Shell scripting with more example programs
5. **System calls**
 - 5.1. Need for protection, System call – User mode and Kernel mode communication, execution of a system call, system call example program in C, difference between system call and function call, system call examples
 - 5.2. System calls for low level file I/Os – open, creat, close, read, write, lseek, mkdir, rmdir, rename, link, symlink, unlink

5.3. System calls related to process – fork, wait, waitpid, exec, signal, kill and raise

6. Assemblers, Linkers and Loaders

6.1. Assemblers – assembler's functions, assembler directives, object code structure, object code, two pass assembler, data structures for two pass assembler – operation code table, symbol table and location counter

6.2. Linker and Loader – Absolute loader, absolute loader implementation. Relocation loader using modification record, Relocation loader using relocation bit mask

6.3. Linker and Loader – Program linking, implementation of linking loader, data structures, external symbol table, program load address and control section address, algorithm. Dynamic linking – dynamic linking applications, dynamic linking implementation

Learning Outcomes:

No	Learning Outcomes
LO1	Students will be able to explain how the Unix command shell processes commands
LO2	Write simple shell scripts
LO3	Explain how the Unix file system stores information
LO4	Explain how concurrent processes are used in Unix
LO5	Basic understanding of assemblers, linkers and loaders

Part B: Contact Session Plan

Academic Term	
Course Title	Systems Programming
Course No	
Lead Instructor	

Course Contents

Contact Hour	List of Topic Title (from content structure in Part A)	Topic # (from content structure in Part A)	Text/Ref Book/external resource
1	Directories and Files in UNIX / Linux system. Basic UNIX commands Getting Introduction to UNIX / Linux	1.1	Ch1(T1),Ch1(R1)

2	Logging in to Linux system	1.2	Ch1(T1),Ch1(R1), Ch1(R5)
3	UNIX files and directories	1.3	Ch2(T1),Ch1(R5)
4	Basic UNIX commands	1.4	Ch1(T1)
5	Internals of Unix file system, I/O in UNIX Basics of files, UNIX file system, Inode structure in UFS, Links in UNIX	2.1	Ch2(T1),Ch4(R6)
6			
7	Directories in UNIX, Conversion of a path name to an inode, open files and descriptor management	2.2	Ch2(T1),Ch4(R6)
8			
9	File system layout, Superblock, I/O in UNIX – Kernel I/O structure, I/O devices – Block and Character devices, Device drivers, I/O queuing and interrupt handling	2.3	Ch2(T1),Ch4(R6)
10			
11	VI editor, File and directory commands, I/O indirection and Filters Working with VI editor: Basic operating modes	3.1	Ch4(R1),Ch1,2,3, 4(R3)
12	Commands for copying & pasting text and pattern searching & substitution of text in command mode, Recovering files in vi editor	3.2	Ch4(R1),Ch1,2,3, 4(R3)
13	File and directory commands – cat, cp, mv, rm, chmod, cmp, diff and top	3.3	Ch5,6,7,9(R1)
14	I/O indirection, Filters – wc, sort, head, tail, grep, pipe and tee	3.4	Ch3(T1)
15	Review		
16	Review		
17	Shell Scripting Pattern matching and Shell Meta characters	4.1	Ch3(T1),Ch2(R4)
18	Shell Scripting: simple programs, interactive shell scripting, arithmetic in shell, variables in shell with more example programs	4.2	Ch3(T1),Ch2(R4)
19	Decision making constructs in Shell with more example programs	4.3	Ch5(T1),Ch3,4,5,6 (R4) Ch8(R2)
20	Loop constructs in Shell with more example programs	4.4	Ch5(T1),Ch3,4,5,6 (R4) Ch8(R2)
21	Files using Shell scripting with more example programs	4.5	Ch5(T1),Ch6(T1), Ch7(R4)
22			
23	System calls Need for protection, System call – User mode and Kernel mode communication	5.1	Ch7(T1),Ch5(R6)
24	execution of a system call, system call example program in C, difference between system call and function call, system call examples	5.1	Ch7(T1),Ch5(R6)
25	System calls for low level file I/Os – open, creat, close, read, write, lseek, mkdir, rmdir, rename, link,	5.2	Ch7(T1),Ch3,4,5(R5),Ch5(R6)

	symlink, unlink		
26	System calls related to process – fork, wait, waitpid, exec, signal, kill and raise	5.3	Ch7(T1)
27	Assemblers, Linkers and Loaders Assemblers – assembler's functions, assembler directives, object code structure, object code, two pass assembler, data structures for two pass assembler – operation code table, symbol table and location counter	6.1	Ch2(R7)
28			
29	Linker and Loader – Absolute loader, absolute loader implementation. Relocation loader using modification record, Relocation loader using relocation bit mask	6.2	Ch3(R7)
30	Linker and Loader – Program linking, implementation of linking loader, data structures, external symbol table, program load address and control section address, algorithm. Dynamic linking – dynamic linking applications, dynamic linking implementation	6.3	Ch3(R7)
31	Review		
32	Review		

Detailed Plan for Lab work/Design work:

1. Any Unix or Linux based system with GCC compiler
2. Can also make use of Virtual lab server for doing labs

Lab No	Lab Objective	Lab Sheet Access URL	Content Reference
1	Execute all the Basic UNIX commands with different options in server and observe the result Basic UNIX commands – passwd, date, who, who am i, mail, write, man, pwd and ls Basic UNIX commands – mkdir, cd [using relative and absolute path], creating a new file using editor ed and command cat, displaying the content of a file using cat and pr commands		Module 1
2	Discussion about various types of links [hard, symbolic] in UFS. Different between these links. Discussion on choosing hard link Vs Symbolic link		Module 2
3	Explore VI Editor and execute all the UNIX commands discussed in this module with different options in server and observe the result		Module 3
4	Execute shell script programs in server Example programs using variables and decision making constructs Example programs using decision making constructs. Example programs using loop construct Files using shell scripting – programs to take data from file, manipulate the data and store it in file		Module 4
5	Write C programs which uses all the system calls directly		Module 5
6	System calls for low level file I/Os – open, creat, close, read, write, lseek, mkdir, rmdir, rename, link, symlink, unlink		Module 5

7	System calls related to process – fork, wait, waitpid, exec, signal, kill and raise		Module 5
8	Demo of Assembler, linker and loader (8086) TASM, TLINK		Module 6
Lab Q's	Please refer to System Programming Lab		

Evaluation Scheme

Evaluation Component	Name (Quiz, Lab, Project, Midterm exam, End semester exam, etc)	Type (Open book, Closed book, Online, etc.)	Weight	Duration	Day, Date, Session, Time
EC - 1	Assignment (Project Presentation)	Contact	20%	25min	To be announced
EC - 2	Mid-term Exam	Closed book	30%	2 hours	To be announced
EC - 3	End Semester Exam	Open book	50%	3 hours	To be announced

Note - Evaluation components can be tailored depending on the proposed model.

Important Information

Syllabus for Mid-Semester Test (Closed Book): Topics in Weeks 1-7

Syllabus for Comprehensive Exam (Open Book): All topics given in plan of study

Evaluation Guidelines:

1. EC-1 consists of either two Assignments or three Quizzes. Announcements regarding the same will be made in a timely manner.
2. For Closed Book tests: No books or reference material of any kind will be permitted. Laptops/Mobiles of any kind are not allowed. Exchange of any material is not allowed.
3. For Open Book exams: Use of prescribed and reference text books, in original (not photocopies) is permitted. Class notes/slides as reference material in filed or bound form is permitted. However, loose sheets of paper will not be allowed. Use of calculators is permitted in all exams. Laptops/Mobiles of any kind are not allowed. Exchange of any material is not allowed.
4. If a student is unable to appear for the Regular Test/Exam due to genuine exigencies, the student should follow the procedure to apply for the Make-Up Test/Exam. The genuineness of the reason for absence in the Regular Exam shall be assessed prior to giving permission to appear for the Make-up Exam. Make-Up Test/Exam will be conducted only at selected exam centres on the dates to be announced later.

It shall be the responsibility of the individual student to be regular in maintaining the self-study schedule as given in the course handout, attend the lectures, and take all the prescribed evaluation components such as Assignment/Quiz, Mid-Semester Test and Comprehensive Exam according to the evaluation scheme provided in the handout.