



**NAGARJUNA COLLEGE OF ENGINEERING & TECHNOLOGY**  
**(An Autonomous under VTU)**  
**DEPARTMENT OF INFORMATION SCIENCE & ENGG.**  
**3<sup>RD</sup> Semester 2019-2020**  
**COURSE HANDOUT**

**Course Code** : 17IST61  
**Course Title** : Unix System Programming  
**Course Teachers** : Prof. Kishore P J  
**Course coordinator** : Prof. Kishore P J

**1. COURSE DESCRIPTION:**

This Course covers the fundamental principles and basics of unix system programming. The main topics covered are unix,ansi and posix standards.unix files and unix file API's .Unix process and control,Process relationship and signals,Inter process communication

**2. COURSE OBJECTIVE:**

This course will enable students to Understand UNIX,ANSI and POSIX standards.

- Write various programs to manipulate files and their attributes.
- Understand process environment ,process control and process relation
- Understand the concept of signal handling and daemon process and its characteristics
- Analyze interprocess communication concepts pipes,message queues,shared memory ,semaphores

**3. COURSE PLAN:**

Class SI No	Module and Title/page no	Topics to be covered	% of portions covered	
			Covered in the chapter	Cumulative
1.	<b>Module I</b>  <b>Introduction T1 page no1-19 And 125-137</b>	UNIX and ANSI standards	20%	20%
2.		Difference between ANSI and C++		
3.		The POSIX standards,the POSIX.1 FIPS standard		
4.		UNIX and POSIX APIs		
5.		File types ,The UNIX and POSIX file system		
6.		The UNIX and POSIX file attributes		
7.		Inodes in UNIX system V		

8.		Application program interface to files		
9.	<b>Module II UNIX files and UNIX file API's T1 page no139-188</b>	UNIX Kernel support for files	20%	40%
10.		Relationship of C stream pointers and file descriptors		
11.		Directory files hard and symbolic links		
12.		General file API's		
13.		File and record locking		
14.		Directory file API's		
15.		Device file API's		
16. O f		FIFO file API's , symbolic file API's		
17.	<b>Module III UNIX process and control T2 page no179-237</b>	The environment of unix process introduction	20%	60%
18.		Main function ,process termination Environment list		
19.		Memory layout of a C program		
20.		Memory allocation,environment variables		
21.		Setjmp and longjmp functions getrlimit ,setrlimit functions		
22.		Unix kernel support for processes		
23.		Fork ,vfork,exit,wait,waitpid ,wait3 ,wait4 functions		
24.		Race conditions,exec functions		
25.	<b>Module IV Process relationship and signals T2 page no242-349</b>	Interpreter files,system function	20%	80%
26.		Process accounting ,user identification Process times		
27.		Terminal logins,network logins		
28.		Process groups,sessions , controlling terminal		
29.		Tcgetpgrp and tcsetgrp functions,job control		
30.		Signal,signal mask ,sigaction		
31.		The sighld signal and waitpid function		
32.		Kill,alarm		
33.		Introduction ,pipes,popen and pclose functions	20%	100%

34.	<b>Module V</b>  <b>Inter-Process communication</b>  <b>T2 page no495-548 and 423-428</b>	Coprocesses , FIFO's		
35.		Message queues and semaphores		
36.		Shared memory		
37.		Daemon characteristics		
38.		Coding rules		
39.		Basics of socket programming		
40.		Basics of socket programming		

#### 4. TEXT BOOK:

**T1** Terrance Chan :”UNIX system programming Using C++”Prentice Hall India,1999,

ISBN -10:0133315622 ,(Listed topics only from chapters 1,5-10)

**T2** W.Richard Stevens:”Advanced programming in the UNIX environment”,2<sup>nd</sup> Edition,Pearson education ,2005,ISBN :0201433079(listed topics only from chapters 7,8,9,13,14,15)

#### 5. REFERENCE BOOKS:

**R1.** Marc J Rochkind : “Advanced UNIX programming “,2<sup>nd</sup> edition ,Pearson education,2005

**R2.** Maurice J Bach: “The design of the UNIX operating system”,Pearson education ,1987

#### 6. EVALUATION SCHEME:

Component	Weightage	Date
CIE 1	20%	
CIE 2	20%	
Makeup CIE	20%	
AAT-1 (Quiz)	5%	
AAT 2 (Surprise Test)	5%	
SEE	50%	

#### 7. COURSE OUTCOMES:

On successful completion of this module, students should be able to:

- Analyze the different manifested constants to determine the system is POSIX conforming
- Differentiate the ANSI C and POSIX standard
- Use API's to implement interprocess communication and other basic services of UNIX kernel.
- Implement fork,race condition , zombie process etc
- Apply system calls to create processes that manipulate system resources and get the system configuration limits

**Course Teacher**  
Mr. Kishore P J

**HOD**  
Dr.Mamatha G