CS-220 – Operating Systems

Form number	COURSE OUTLINE / DOCUMENT	
COURSE INSTRUCTOR INFORMATION	Name	Dr. Rana Asif Rehman
	Email ID	r.asif@nu.edu.pk

DEGREE INFORMATION	Program	Batch	Section(s)			Semester	Spring
	BSCS	2019	4B	4C		Year	2021

COURSE INFORMATION	Course Category C- Core/ E-Elective	Code	Title	Credit hours
	C	CS-220	Operating Systems	3 + 1
	Prerequisite(s)	CS-201	Data Structures	
	TA Required (Yes/ No)	No. of TA(s)	Brief Justification	
	Yes	1		

TEXT BOOK(s) INFORMATION			
	Title of Book	*Operating System Concepts*	Edition: 9th or 10th
	Author(s)	*Abraham Silberschatz et al.*	
	Publisher	*Prentice Hall*	
Reference Book (s)	1. Title of Book	*Modern Operating System, Author (s): Tenenbaum*	
	2. Title of Book	*Operating Systems, Author (s): William Stallings*	
	Support Material(s) a.		
	b.		
	c.		
	d.		

Brief Description of Course:

Operating System is a core course offered to BSCS. This course will cover the theoretical as well as practical knowledge of Operating System concepts. Major topics to be covered are Process Management, Thread Management, Process Synchronization, Deadlock management, Memory Management and File Management. Security and Distributed Operating Systems will be discussed subject to the availability of time. This is a programming intensive course and involves implementation and use of system calls, multithreaded applications, simulation of memory management, scheduling etc. with a lot of emphasis on synchronization issues.

Course Objectives (CO):	
1.	Understanding components of an operating system
2.	Learn how the Operating systems evolved
3.	Learning thread programming and memory management
4.	Understanding the problems that are faced in process execution when shared resources are involved
5.	Develop understanding of organization and structure of file system
6.	Providing the knowledge for algorithms and data structures used in development of an OS.

Learning Outcome (LO):	
1.	On the successful completion of this course, students should be able to understand basic operating system concepts
2.	The students should be able to understand different aspects of concurrency control and memory management
3.	On the course completion, students should have ability to implement threads using a programming language
4.	The students should have ability to understand the scheduling algorithms implemented in OS
5.	The students will be capable to understand and implement file system

Outcome Assessments	Outcomes are assessed through quizzes, assignments sessional I, II and final examination. These assessments are conducted throughout the whole semester.
Technologies	This is an introductory course to Operating Systems. The technologies used are Linux programming, windows programming and multi-threading.
Design skills/ Techniques	Common techniques and algorithms used in operating system design are discussed and students have to implement and make running model of the algorithms. The creation of simulation involves software and OS design skills.
Effectiveness of course and level of skills	This course provides foundation of multi-threaded programming. Also provides in-depth knowledge how multi-processing is achieved and how to exploit the hardware specifications to get the best out of applications. Complete lifecycle of a processing from start to end of execution is discussed in detail.
Emerging paradigms	Embedded operating systems strive to get the best of specialized hardware utilizing minimum resources to manage the specialized task. As part of this course students had to configure and execute **TinyOS** one of emerging embedded OS.
Modeling and Design	As part of the course students are made to design and implement subparts of OS.

Courseware Structure:

Lecture (Lect)	*Multimedia (MM)*	*Exercise (s) (Exer)*	*Labs (Lab)*	*Case Studies (CAS)*	*Assignment (s) (Assign)*	*Group Presentation (G-Pres)*	*Any other Medium*
X	X	X	X	X	X		

COURSE CONTENTS:			
Weeks	**Contents/Topics**	**Lab Activities**	**Evaluation Artifact**
Week-01	Introduction, Overview & Organization	Ubuntu(Linux) installation on VMWare, USB and on Core	
Week-02	Interrupts & Processes	Basic Programing and Grub Editing OS boot process (OS, MBR, Grub, Kernel, Init, Runtime processes)	Assignment_1, Quiz_1
Week-03	PCB, Process Creation & Fork Semantics	Editing grub, GCC compiler and VIM editor,Writing basic C/C++ programs, Command line arguments	
Week-04	IPC, Pipes & Signals	Linux Commands, NASM compiler, stat function	Assignment_2,
Week-05	CPU Scheduling	Commands, Switches, Shell Scripting (Basic Scripting and Variables), Loops,	Quiz_2
Week-06	Threads & Posix Threads	----------	Mid Term_1
Week-07	Deadlocks	Advanced shell scripting, arrays, functions	
Week-08	Synchronization, Busy Waiting &Bakery Algorithms	Bootloader Develompemt, Virtual floppy, Qemu Emulator	Assignment_3, Quiz_3
Week-09	TSL, Priority Inversion, Semaphores	Kernel Modules (List, Description, Add, Remove)	
Week-10	Semaphores, Readers Writers, Dining Philosophers	System Calls, fork(), excel, getpid(),getgrp() etc.	Quiz_4
Week-11	Revision, Mid term 2	----------	Mid Term_2
Week-12	Memory Management & its Data Structure	Inter process communication using Pipes , read, write, dup2() etc.	
Week-13	Paging & Segmentation, Memory Management Policies	Threads , create ,exit, join etc	Assignment_4, Quiz_5
Week 14	File System Interface & Implementation	Synchronizations	
Week-15	OS Case Studies; Recaps,	Kernel Recompilation	

	Presentations, Discussions		
Week-16	Revision Week and Problem solving week	Final Lab	
Week-17	Final Term Exam		Final Exam

Grading Criteria			
	ABSOLUTE Grading	X	RELATIVE Grading

Marks Distribution:

Particulars	% Marks
1. Assignments	10
2. Quizzes	15
4. Mid Term 1	15
5. Mid Term 2	15
6. Final Exam	45
Total:-	**100**

Planned Courseware Events:

Particulars	Planned (Qz/As/Labs)	Remarks
1. Quizzes	**5-6**	Announced and unannounced
2. Assignments	**4**	Individual assignments

QUALIFYING ATTENDANCE	You must attend every class for your own personal benefit. Please refer to university policy of minimum attendance requirement (currently 6 absentees – 1.30 hrs (90 minutes) each class duration). Failing to conform qualifying attendance threshold, the student will stand debarred from sitting in the examination and assigned with "F" Grade.
Academic and Moral Integrity:	1. All assignments should be your own work (or your group's when approved). PLAGIARISM will be awarded with "F" grade and/or reported to the University for academic and moral misconduct. To protect yourself, ALWAYS PROVIDE REFERENCES! 2. **Missed quizzes/presentations/assignments** will not be rescheduled. 3. **Late/Copied assignments** shall not be accepted and will result in deduction of marks already scored.

Instructions / Suggestions for STUDENTS for satisfactory progress in this course:

- ✓ On average, most students find at least three hours outside of class for each class hour necessary for satisfactory learning.
- ✓ Chapters should be read and homework should be attempted before class.
- ✓ You may contact me through email on **r.asif@nu.edu.pk**.
- ✓ The homework assigned is a minimum. You should always work extra hours on your own.
- ✓ Use the few minutes you usually have before the start of each class to review the prior meetings' notes and homework. This will save us valuable in-class time to work on new material.
- ✓ Develop a learning habit rather than memorizing; work in groups, whenever appropriate.
- ✓ Apply the learned principles and gained knowledge; be creative in thinking.
- ✓ **Assignments/ Activities:** They are not meant simply for grades, but to reinforce your learning. Assignments are due on time. Each day late will lower your assignment grade by 10%. Apart from value of content, spelling, grammar, punctuation, and good presentation (printing and paper quality) will figure into your assignment grade. To guard against errors, please keep copies of the papers you turn in and retain all graded assignments for your reference.
  Your Assignments must include all the References. For this course you are highly encouraged to follow the Harvard style of referencing