CSC 343 – Operating Systems, Spring 2021 First Day Handout

http://faculty.kutztown.edu/parson

Section .810: TuTh 12-1:20; section .811: TuTh 1:30-2:50, Zoom classes & recordings,
Class-time Zoom for CSC343: See D2L Course CSC343 -> Content -> Overview for the link.
IF you don't want to be recorded or are a minor, use PRIVATE ZOOM CHAT to me for questions.
Dr. Dale E. Parson, parson@kutztown.edu, Office hours: https://kutztown.zoom.us/j/94322223872
Office Hours Monday 2-4, Wednesday 1-3, Thursday 10-11 or by appt.

This course is an introduction to the basic software components of an operating system. Topics include process management, storage management, processor management, auxiliary storage management, evaluation of an operating system's performance, networks, operating system security, case studies of particular operating systems.

Prerequisite: CSC 237 or CSC 253 AND completion of 18 credits of CSC courses numbered 125 or higher AND a GPA in the CSC courses of 2.25.

Recommended Textbook:

Operating System Concepts, 10th Edition, Silberschatz, et. al. (9th edition is OK)

OR *Operating System Concepts Essentials*, 2nd Edition, Silberschatz, et. al.

(9th edition) & *Essentials Edition*. Chapter numbers vary.

Grading (A = 92:100, A- = 90:91, B+ = 87:89, B = 82:86, B- = 80:81, C+ = 77:79, C = 70:76, D = 60:69, F = 0:59)

Projects 100% divided equally among the project assignments.

Programming project assignment grading criteria

Grading rubrics will be part of each assignment handout. Late penalty is 10% per each day late, up until I go over the solution. Any assignment turned in after that is worth 0%.

We will use the CS&IT documentation requirements:

http://faculty.kutztown.edu/parson/CSCDocumentationStandards.pdf

The academic integrity policy:

http://faculty.kutztown.edu/parson/AcademicIntegrityPolicy.pdf

Please read the above policy statement.

You may openly discuss ideas, algorithms, pitfalls, and the use of programming tools.

You may not share code, test drivers or test data except within groups for group projects.

Class attendance is not graded, but I will be teaching using data sources and concepts both inside and outside the scope of the textbook. You are responsible for all material covered in class, including technical information, coding standards and conventions, verbal specification of assignments, and your questions about topics that are not clear to you. Please, there should be no classroom conversations, cell phones, text messaging, eating, sleeping, obscenities, listening to music or other disruptions of the class.

If you have already disclosed a disability to the Disability Services Office (215 Stratton Administration Building) and are seeking accommodations, please feel free to speak with me privately so that I may assist you. If you have an injury sustained during military service including PTSD or TBI, you are also eligible for accommodations under the ADA and should contact the Disability Services Office.

If you have preferred pronouns for yourself, or a name that differs from the MyKU roster, please let me know.

Week	Text chapters	(Essentials Chapter / 10th Edition Chapter) Lecture Topics
1	1	Introduction, computer architecture, hardware / software interface, history
2	2, P[1]	State-machine based simulation architecture in Python.
3	3	Processes, inter-process communication (IPC), modeling using state diagrams
4	4, 5 (10th 4, 6) P[2]	Threads, thread libraries, queuing, related state machine models
5	5,6(10th 5,6)	Process synchronization, CPU scheduling
6	6 (10th 7,8ish)	CPU scheduling, deadlocks
7	7 (10th 9), P[3]	Main memory, memory hierarchies, swapping, paging
8	8 (10th 10)	Review.
9	8 (10th 10)	Virtual memory, scheduling memory
10	9,10(10th 11,13)P[4]	Mass storage systems, File system interface
11	11 (10th 14)	File system implementation, directories, allocation / free space management, NFS
12	12 (10th 12)	Disk scheduling, swap space management, RAID
13	12 (10th 12)	Input-output (I/O) systems, buses and multiprocessor interconnectivity
14	P[5]	Preparation for final exam project.
15		Final project work session.

There will be 5 project assignments and no exams. A compressed, 1-week review assignment 5 takes the place of an exam.

[1] Project 1 is a practice project for learning how to construct, compile and simulate state machines based on the UML (Unified Modeling language) State Diagram notation. **Following project sequence is a draft plan**.

[2] Project 2 involves building state machine models to simulate various CPU scheduling algorithms.

[3] Project 3 involves building interacting state machines to share resources and avoid / risk deadlock.

[4] Project 4 involves building a simulation model to simulate memory mapping, swapping and page scheduling.

[5] A compressed, 1-week review assignment 5 takes the place of an exam.