Computer Science Department
cs.salemstate.edu

SYLLABUS **Fall, 2020**

CSC 381 Operating System Principles **3.0 credits**
Prerequisite(s): CSC 260

Instructor: Beifang Yi **Office**: MH 211A/Online/Zoom **Phone**: (978) 542-7246/Zoom
email: byi@salemstate.edu **Hours**: TR (10:40am-12:45pm) **Web Site**: http://cs6.salemstate.edu/~byi/

Section	Time	Room	Final Exam
01	T & R 9:25am—10:40am	Online	**Dec 18, Friday, 8:00am—10:00am Online**

(SSU Course Online is at: **https://elearning.salemstate.edu** **)**

Detailed Course Offering Information through Zoom

Time	ZOOM	Final Exam
Tuesdays/Thursdays 9:25am—10:40am	ZOOM Meeting ID: **998 7786 1034** URL: **https://salemstate.zoom.us/j/99877861034** One tap mobile: +13126266799,,99877861034#,,,,,,0#,,054142# US (Chicago) +19292056099,,99877861034#,,,,,,0#,,054142# US (New York)	**Dec 18, Friday, 8:00am—10:00am** **Online** https://salemstate.instructure.com/
Note: due to the Zoom scheduling specifics, Zoom schedules our class formally starting **at 9:30am**, but you may login at 9:25am or earlier.		

Detailed Office Hours through Zoom

Days/Times	ZOOM (URL)	ZOOM (Mobile)
Tuesdays/Thursdays **10:40am-12:45pm**	Meeting ID: **965 3437 5678** **https://salemstate.zoom.us/j/96534375678**	+13017158592,,96534375678#,,,,,,0#,,854359# US (Germantown) +13126266799,,96534375678#,,,,,,0#,,854359# US (Chicago)
Note: due to the Zoom scheduling specifics, Zoom schedules my office hours formally starting **at 11:00am**, but you may login at 10:45am or earlier; also note that this Zoom meeting is ***different*** from our class meeting so you may need to log in via this Office Hour Meeting ID.		

Catalog description:

This course presents the evolution of computer operating systems, operating system functionalities, and current design and implementation techniques. Relationships between the operating system, computer architecture, and the user community are discussed. Three lecture hours per week.

Course Goals:

The aims of this course are:

- CG1: present a descriptive overview of modern operating systems, their purposes and design principles;
- CG2: discuss the most important ingredients, techniques, and algorithms used in their construction;
- CG3: foster an understanding of fundamental technical issues in the implementation of modern operating systems.

Course Objectives:

Upon completion of this course, the student will have demonstrated the ability to:

- CO01: summarize the development of operating systems from single-user and batch processing mainframes to modern multitasking systems
- CO02: describe the mechanisms of interrupts and Direct Memory Access;
- CO03: describe how a process or task is represented in a modern computer system;
- CO04: differentiate between the concepts of process and thread and describe the behavior of a multithreaded system;
- CO05: describe the system components, actions, and algorithms involved in scheduling and managing concurrent processes;
- CO06: describe the concept of deadlock and the common techniques for recognizing, predicting, avoiding, and recovering from it;
- CO07: describe the common techniques and problems involved in memory management, including paging and virtual memory;
- CO08: describe the common techniques and problems involved in management of disk storage;
- CO09: describe the common techniques and problems involved in file management;
- CO10: describe the common techniques and problems involved in system protection and security.

Course Topics:

A detailed topics list and a general course bibliography can be found on the Computer Science Department website at https://cs.salemstate.edu/courses/course-information. Select CSC 381 Operating System Principles to access a PDF document.

The focus of the course is on a general discussion of the nature and functionality of operating systems. There is no extended treatment of specific case studies, although examples drawn from specific systems are used to illustrate major concepts. In addition to the topics listed in the above link, we may also briefly introduce the following special purpose systems:

- threads and thread programming
- virtualization/virtual machines
- Linux

Text(s): (required) **Operating System Concepts**, **10th** Edition, by A. Silberschatz, G. Gagne, et al. Wiley, 2018.
(**Note:** get the text's ***10th edition***, ***electronic version E-book*** is preferred which provides some interactive learning materials including quick quiz questions and solutions.)

Required Material:

(Required) Thumb (flash) drive (or other portable memory devices and/or online storage), 16 GB minimum**.**

Additional references (optional)**:**

- Couse website: http://cs6.salemstate.edu/~byi/CSC381_Silberschatz/index.html
- *Linux with Operating System Concepts*, by Richard Fox, Chapman & Hall/CRC, 2017.
- *Operating Systems: Three Easy Pieces*, 1st Edition, by CreateSpace Independent Publishing Platform, 2018.
  - http://pages.cs.wisc.edu/~remzi/OSTEP/
- *Mastering Lambdas: Java Programming in a Multicore World* (Oracle Press), 1st Edition, by Maurice Naftalin. McGraw-Hill Osborne Media, 2014.
  - https://docs.oracle.com/javase/tutorial/java/javaOO/lambdaexpressions.html
  - http://www.oracle.com/webfolder/technetwork/tutorials/obe/java/Lambda-QuickStart/index.html
- *Start Concurrent: An Introduction to Problem Solving in Java with a Focus on Concurrency*, by Barry Wittman. And Tim Korb. Purdue University Press (Kindle/2014, paperback/2013). Or through the following link:
  - https://start-concurrent.github.io/

- *Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming*, 1st edition, by Simon Marlow. O'Reilly Media, 2013.
- *Java: How to Program*, 11th edition. Deitel & Deitel. Prentice Hall, 2018.
- *Effective Java*, 3rd Edition, Addison-Wesley Professional, 2018.
- VMWare website: https://www.vmware.com/.
- (Other course study guides and materials will be posted on the course website and/or LMS/Canvas.)

Cell phones:

Turn the ringer off, or, better yet, *turn the phone off*.

Class Attendance:

Class attendance is highly recommended. You are responsible for all materials presented in class, examinations, and other announcements. No excuses of any nature will be construed as relieving you from the responsibility for completion of the work assigned. Each student is responsible for completing all course requirements and for keeping up with all that goes on in the course (whether or not the student is present).

Final Grade:

Final grade will be determined using the following grading weights:

written assignments	20%
projects	30%
mini-tests	20%
midterm examination	10%
final examination	20%

Attendance is not used to calculate the final overall grade: *however*, note that you are at all times responsible for all the assignments and materials presented in class.

The following table shows how the course work is assessed against the course objectives:

	Written Assignments	Projects	Examinations
CO01	✓		✓
CO02	✓		✓
CO03	✓	✓	✓
CO04	✓	✓	✓
CO05	✓	✓	✓
CO06	✓	✓	✓
CO07	✓	✓	✓
CO08	✓	✓	✓
CO09	✓		✓
CO10	✓		✓

Projects:

There will be several projects (including programming projects) to be completed throughout the semester. The programming languages and platforms will vary (C/C++, Java…Windows, Linux…) and will be given out in class. These projects will help students understand the basic concepts of modern operating systems and get hands-on experience in the implementation of some concepts and functions of the operating systems. There is a deadline to each project and ***penalty*** will be imposed for late submissions (see the Due Dates/Time below).

For each project, a write-up should be submitted which describes the problem, solutions, and other requirements (usually, there are additional requirements for the assignments). For a programming project, it should provide the following: (a) instructions on how to compile and run the program, (b) testing results, (b) source code, and (d) input data (if applicable).

Please note that each project has specific requirements and it is important that you read these requirements before working on the project.

All the projects must be submitted at Canvas (https://elearning.salemstate.edu/).

Homework/Written Assignments:

There will be a series of written assignments from the textbook and other sources: question-answering, short essay-writing, and/or reporting of calculation results. Reading will be a part of the written assignments. Please note that in addition to these (written) assignments there will be (Programming) Projects. There is a deadline to each assignment and ***penalty*** will be imposed for late submissions (see the Due Dates/Time below).

All the assignments must be submitted at Canvas (https://elearning.salemstate.edu/).

Each assignment may have different full score points, depending on the difficulty and the amount of the work of the exercises. All the assignment questions might be in the examinations.

Mini-tests are the extended course assignments that are closely related with the written assignments. They serve as an important study aid and testing procedures and evaluations of the learning of coursework: first working on the assignments and then taking these mini-tests for the enhancement of the understanding of the course subjects.

Examinations/Tests/Mini-tests:

There will be one midterm and one final (comprehensive) examination. The midterm will be held in the "middle" of the semester *depending* on class progress. **Note:** Make-ups are given for examinations only under exceptional circumstances and with documented circumstances (electronic documents are OK under special circumstances).

Mini-tests serve as important testing procedures and evaluations of the learning of coursework: first working on the assignments and then taking these mini-tests for the enhancement of the understanding of the course subjects.

Missed Tests/Mini-tests/Examinations:

Missed tests will be made up *only under extreme conditions/emergency with the proper documentation*. Students who know in advance that they must be absent on an exam day for an excusable reason should notify the instructor prior to the exam day. Students who are absent on the day of the exam for an excusable reason should contact the instructor immediately following their absence. Makeup work will be permitted *only when* the instructor is presented with acceptable documentation for acceptable absences. It is student's responsibility to notify the instructor of any excused absence as far in advance as possible.

Due Dates/Time:

- Late submission of assessed assignments or projects will result in penalties defined in the following:
  - **25 percentage** points will be deducted for being late, up to 24 hours.
  - **50 percentage** points will be deducted for being late, more than 24 hours and up to 48 hours.
  - **75 percentage** points will be deducted for being late, more than 48 and up to 72 hours.
  - **100 percentage** points will be deducted for being late more than 72 hours (i.e., a grade of zero will be given).
  - A grade of **zero** will be given for any assignments including projects which are submitted after the final examination time. That is: **no assignments (including semester project) will be accepted after the final examination**.
- Should there be an emergency that prevents you from completing/submitting your assignments/projects on time, you will need to send *an email request* for the extension on the coursework submission. The instructor will reply to this request email with a specific number of days for the extension period or a new deadline for the assignment and you will need to keep *this email as a record of the extension approval*. Sending only a request email does ***not*** guarantee the extension approval
- **Please double-check** your submissions (since your assignment submissions are usually graded after their deadlines): to guarantee your successful and correct submissions, you would need to download your submissions and examine the downloaded materials; as for the projects, you would need to check the downloaded programs and then compile/run the code.

Study Groups:

While I strongly encourage study groups, I require that each student hand in his/her answers in her/his own words - if two answers

are highly similar to each other, neither will receive credit.

When working on your programming projects, you may discuss with others the project topics, the algorithms and methodologies related to the project; but when you work on writing the code, this coding work should be 100% of your own work. **If two answers/written code segments come out exactly the same or highly similar, neither will receive credit and/or further actions will be taken** (such as reporting to the department and/or university).

Academic Integrity:

Academic Integrity Policy and Regulations can be found in the University Catalog and on the University's website (http://catalog.salemstate.edu/content.php?catoid=13&navoid=1295#Academic_Integrity). The formal regulations are extensive and detailed - familiarize yourself with them if you have not previously done so. A concise summary of and direct quote from the regulations: "Materials (written or otherwise) submitted to fulfill academic requirements must represent a student's own efforts". *Submission of other's work as one's own without proper attribution is in direct violation of the University's Policy* and will be dealt with according to the University's formal Procedures.

All students are expected to be familiar with the academic regulations, including those regarding Academic Integrity, for Salem State University as published in the college catalog. In addition, each student is responsible for completing all course requirements and for keeping up with all that goes on in the course (whether or not the student is present).

Salem State University is committed to providing equal access to the educational experience for all students in compliance with Section 504 of The Rehabilitation Act and The Americans with Disabilities Act and to providing all reasonable academic accommodations, aids and adjustments. Any student who has a documented disability requiring an accommodation, aid or adjustment should speak with the instructor immediately. Students with Disabilities who have not previously done so should provide documentation to and schedule an appointment with Disability Services and obtain appropriate services.

Students must comply with the Covid-19 Health and Safety Protocols for the 2020-2021 Academic Year. This includes wearing masks in class and on campus in public spaces, practicing physical distancing where possible, including in class, engaging in a daily symptom check, notifying Counseling and Health Services at 978-542-6413 if they have any symptoms associated with COVID-19, and not coming to campus or to an in-person class if they have any of the symptoms related to COVID-19, until cleared by the Student Life Wellness Area. Students who have documented disabilities that may prevent them from complying with these policies are required to contact the Disability Services office.

In the event of a university declared critical emergency, Salem State University reserves the right to alter this course plan. Students should refer to Salem State for further information and updates. The course attendance policy stays in effect until there is a university declared critical emergency. In the event of an emergency, please refer to the alternative educational plans for this course located at Canvas (https://elearning.salemstate.edu/). at/in [faculty member determines this]. Students should review the plans and gather all required materials before an emergency is declared.

Please remember that if, for any reason, you decide to drop this course, you **MUST** do so officially through the Registrar's office. The last day to withdraw from a course this semester is **November 20th**.

Note: This syllabus represents the intended structure of the course for the semester. If changes are necessary, students will be notified in writing and via all regular class communication mechanisms (class discussion, emails, and/or the course link at Canvas https://elearning.salemstate.edu).

Course Schedule for CSC381
(Subject to Change)

Week	Dates	Contents (textbook chapters and others)
1	9/2—9/4	Warming-up: Windows, Unix, Linux, Virtual Machines—Ch. 18
2	9/7—9/11	Computer System/Organization Ch. 1 Introduction
3	9/14—9/18	Ch. 1 Introduction Ch. 2 OS Structures
4	9/21—9/25	Ch. 3 Processes Ch. 4 Thread & Concurrency
5	9/28—10/2	Ch. 4 Thread & Concurrency Ch. 5 CPU Scheduling
6	10/5—10/9	Ch. 5 CPU Scheduling Ch. 6 Synchronization Tools
7	10/12—10/16	Java Thread Programming Ch. 7 Synchronization Examples
8	10/19—10/23	Ch. 8 Deadlocks
9	10/26—10/30	Ch. 9 Mai Memory Ch 10 Virtual Memory
10	11/2—11/6	Ch. 11 Mass-Storage Structure Ch. 12 I/O Systems
11	11/9—11/13	Ch. 13 File-System Interface Ch. 14 File-System Implementation Ch. 15 File-System Internals
12	11/16—11/20	Ch. 16 Security Ch. 17 Protection
13	11/23—11/27	Ch. 18 Virtual Machines **(Thanksgiving Recess)**
14	11/30—12/4	Ch. 19 Networks & Distributed Systems
15	12/7—12/11	Other Topics: Case Studies Reviews
16	12/14—12/18	**Final Examination** Dec 18th, Friday, 8:00am—10:00am, Online