Course Schedule (CAVEAT: mostly set but there could be some updates)

Course Schedule

Course Schedule

Week	Activity	Video Length	Date Due	Notes
1	OS Refresher	2h, 18min	Friday Aug 23	(Optional) Students who need a refresher on AOS topics should take this course
1	Lesson 1: Intro to AOS	46min	Friday Aug 23	
2	Homework		Tuesday Aug 27 (11:59 PM EST)	(1 week)
2	Pre-lab		Tuesday Aug 27 (11:59 PM EST)	(1 week)
2	Lesson 2: OS Structures	2h, 40min	Friday Aug 30	
3	Lesson 3: Virtualization	1h, 53min	Friday Sep 6	
4-5	Lesson 4: Parallel Systems	5h, 34min	Friday Sep 13 / Sep 20	This is the last lesson that will be included in the Test 1.
6	Project 1: Virtual Machine Scheduling in KVM		Monday Sep 23 (11:59 PM EST)	(3 weeks)

Week	Activity	Video Length	Date Due	Notes
6-7	Lesson 5: Distributed Systems	3h, 20min	Friday Sep 27 / Oct 4	
6-7	Test 1		Fri Sep 27 (11:59 PM EST) - Mon Sep 30 (11:59 PM EST)	Lesson 1-4
8	Lesson 6: Distributed Objects and Middleware	1h, 56min	Friday Oct 11	
9	Project 2: Barrier Synchronization		Monday Oct 14 (11:59 PM EST)	(3 weeks)
	Fall Break		Monday Oct 14 - Tuesday Oct 15	
9-10	Lesson 7: Distributed Subsystems	3h, 48min	Friday Oct 18 / Oct 25	October 26th, 4 PM ET is the withdrawal deadline.
11	Lesson 9: Internet Computing	2h, 34min	Friday Nov 1	
12	Project 3: Distributed Service using GRPC		Monday Nov 4 (11:59 PM EST)	(3 weeks)
12	Lesson 10: RT and Multimedia	1h, 15min	Friday Nov 8	
12-13	Test 2		Fri Nov 8 (11: 59 PM EST) - Mon Nov 11 (11: 59 PM EST)	Lessons 5-7
13	Lesson 8: Failures and Recovery	1h, 58min	Friday Nov 15	
14	Lesson 11: Security	1h, 17min	Friday Nov 22	
15	Project 4: Implement MapReduce		Monday Nov 25	(3 weeks)

Week	Activity	Video Length	Date Due	Notes
	Framework		(11:59 PM EST)	
	Thanksgiving Break		Thursday Nov 28 - Friday Nov 29	
17	Test 3 (Final Exam)		Fri Dec 6 (11: 59 PM EST) - Mon Dec 9 (11:59 PM EST)	Lessons 8-11

Reading List

Reading List

Lecture	Papers
Lesson 2: OS Structures	1. Brian Bershad et al., "Extensibility, Safety and Performance in the SPIN Operating System ", Proceedings of the 15th ACM Symposium on Operating System Principles, December 1995. 2. Dawson R. Engler, Frans Kaashoek and James O'Toole, "Exokernel: An Operating System Architecture for Application-Level Resource Management ", Proceedings of the 15th ACM Symposium on Operating System Principles, ACM, December 1995. 3. J. Liedtke, " On Micro-Kernel Construction ", Proceedings of the 15th ACM Symposium on Operating System Principles, ACM, December 1995. 4. J. Liedtke, "Improved Address-Space Switching on Pentium Processors by Transparently Multiplexing User Address Spaces ", GMD Technical Report No. 933, November 1995 (self-study).
Lesson 3: Virtualization	1. Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, Andrew Warfield, "Xen and the Art of Virtualization ", SOSP 2003. 2. Carl Waldspurger, “Memory Resource Management in VMware ESX Server”, OSDI, 200.
Lesson 4: Parallel Systems	1. Mellor-Crummey, J. M. and Scott, M., "Algorithms for Scalable Synchronization on Shared-Memory Multiprocessors ", ACM Transactions on Computer Systems, Feb. 1991. 2. B. N. Bershad, T. E. Anderson, E. D. Lazowska, and H. M. Levy. Lightweight Remote Procedure Call . ACM Transactions on Computer Systems, 8(1):37--55, Feb. 1990. 3. (partial reading: skip system modeling) M.S. Squillante and E.D. Lazowska, " Using Processor-Cache Affinity Information in Shared Memory Multiprocessor

	Scheduling ", IEEE Transactions on Parallel and Distributed Systems, Feb. 1993, pgs. 131-143. 4. Alexandra Fedorova, Margo Seltzer, Christopher Small and Daniel Nussbaum. Performance of Multithreaded Chip Multiprocessors and Implications for Operating System Design. Usenix 05. 5. Ben Gamsa, Orran Krieger, Jonathan Appavoo, and Michael Stumm, Tornado: Maximizing Locality and Concurrency in a Shared Memory Multiprocessor Operating System , 1999 Symposium on Operating System Design and Implementation. 6. (partial reading: Sec 1, 2, 3, and 10) S. Boyd-Wickizer, H. Chen, R. Chen, Y. Mao, F. Kaashoek, et. al, "Corey: An Operating System for Many Cores", OSDI 2008. 7. (partial reading: Sec 1, 2, 3, and 8) Kinshuk Govil, Dan Teodosiu, Yongqiang Huang, and Mendel Rosenblum. Cellular Disco: resource management using virtual clusters on shared-memory multiprocessors. In Proceedings of 17th Symposium on Operating Systems Principles, 1999.
Lesson 5: Distributed Systems	1. Lamport, L., " Time, Clocks, and the Ordering of Events in a Distributed System ", Communications of the ACM, 21, 7, pgs. 558-565, July 1978. 2. C.A. Thekkath and H.M. Levy, " Limits to Low-Latency Communications on High-Speed Networks ", ACM Transactions on Computer Systems, May 1993. 3. Hutchinson N.C., Peterson, L.L., " The x-Kernel: An Architecture for Implementing Network Protocols ", IEEE Transactions on Software Engineering, 17, 1, pgs. 64-76, January 1991. 4. David Wetherall, " Active Networks: Vision and Reality: Lessons from a Capsule-based System ", 17th ACM Symposium on Operating System Principles, OS Review, Volume 33, Number 5, Dec. 1999. 5. Liu, Kreitz, van Renesse, Hickey, Hayden, Birman, Constable, "Building Reliable High Performance Communication Systems from Components ", 17th ACM Symposium on Operating System Principles, OS Review, Volume 33, Number 5, Dec. 1999. 6. (partial reading) Schroeder, M., and Burrows, M., " Performance of the Firefly RPC", Proceedings of the Twelfth ACM Symposium on Operating Systems Principles, pgs. 83- 90, December 1989.
Lesson 6: Distributed Objects and Middleware	1. Mitchell, J. G., et al., " An Overview of the Spring System ", Proceedings of Compcon, Feb. 1994. 2. Hamilton, G., Powell, M.L., and Mitchell, J.J., "Subcontract: A Flexible Base for Distributed Programming ", Proceedings of the Fourteenth ACM SOSP, pgs. 69-79, December 1993. 3. Wollrath, A., Riggs, R., and Waldo, J., "A Distributed Object Model for the Java System ", Usenix Conference on Object Oriented Technologies and Systems, May 1996. 4. Emmanuel Cecchet, Julie Marguerite, Willy Zwaenepoel, "Performance and Scalability of EJB Applications", Proceedings of the 17th ACM SIGPLAN

	conference on Object- oriented programming, systems, languages, and applications.
Lesson 7: Distributed Subsystems	1. Feeley, Morgan, Pighin, Karlin, Levy, Thekkath,, "Implementing Global Memory Management in a Workstation Cluster ", Fifteenth ACM Symposium on Operating System Principles, Dec. 1995. 2. C. Amza, A. Cox, S Dwarkadas, P Keleher, H Lu, R. Rajamony, W. Yu and W. Zwaenepoel, " TreadMarks: Shared Memory Computing on Networks of Workstations " IEEE Computer, February, 1996. 3. Anderson, T. et al., " Serverless Network File System ", ACM Transpaction on Computer Systems, February 1996. 4. (partial reading) Mahadev Satyanarayanan, "Coda: A Highly Available File System for a Distributed Workstation Environment", IEEE Trans. Computers, vol 39, no 4, Apr 1990
Lesson 8: Failures and Recovery	1. Satyanarayanan, M., et al., " Lightweight Recoverable Virtual Memory ", The Proceedings of Fourteenth ACM Symposium on Operating System Principles, pgs. 146- 160, December 1993. 2. David E. Lowell and Peter M. Chen, " Free Transactions With Rio Vista ", Proceedings of the Sixteenth ACM Symposium on Operating System Principles, October 1997. 3. R. Haskin et. al., " Recovery Management in QuickSilver ", ACM Transactions on Computer Systems, February 1988. 4. (read on your own) J. N. Gray, P. McJones, M. W. Blasgen, R. A. Lorie, T. G. Price, G. R. Putzolu, and I. L. Traiger. " The Recovery Manager of a Data Management System ", ACM Computing Surveys, Vol. 13, No. 2, June 1981, pp. 223-242. 5. (partial reading: first 3 sections of the paper) D. Porter, O. Hofmann, C. Rossbach, A. Benn, E. Witchel, “Operating System Transactions“, SOSP’09. 6. (partial reading) D. Peng, F. Dabek, “Large-scale Incremental Processing Using Distributed Transactions and Notifications”, OSDI’10
Lesson 9: Internet Computing	1. Dean, J., and Ghemawat, S. "MapReduce: Simplified Data Processing on Large Clusters". 2. (partial reading) Brewer, E. "Lessons from Giant-Scale Services" . 3. (partial reading) Luis Andre Barroso, Jeffrey Dean, Urs Holzle, " Web Search for a Planet: The Google Cluster Architecture ", IEEE Micro. 4. Freedman, M., Freudenthal, E., and Mazières, D. "Democratizing content publication with Coral" . 5. G. DeCandia, D. Hastorun, et. al., “Dynamo: Amazon’s Highly Available Key-value Store”, SOSP’07. 6. (read on your own for learning about Web Technologies) (2 short papers) 1. Curbera, F., Duftler, M., Khalaf, R., Nagy, W., Mukhi, N., Weerawarana, S., " Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI ", IEEE Internet Computing, Volume: 6 Issue: 2, March-April 2002, pgs. 86 -93.

	2. Curbera, F., Khalaf, R., Mukhi, N., Tai, S., Weerawarana, S., " The Next Step in Web Services ", Communications of the ACM, Volume 46 Issue 10 ,October 2003, pgs. 29-34.
Lesson 10: RT and Multimedia	1. Ashvin Goel, Luca Abeni, Charles Krasic, Jim Snow, Jonathan Walpole, Supporting Time-Sensitive Applications on a Commodity OS, OSDI 2002. 2. T. Broomhead, L. Cremean, J. Ridoux, D. Veitch, “Virtualize Everything but Time”, OSDI’10. 3. David Hilley and Umakishore Ramachandran, Persistent Temporal Streams . ACM/IFIP/USENIX 10th International Middleware Conference, Urbana Champaign, Illinois, USA , November 30 - December 4, 2009. 4. Shahabi, Zimmermann, Fu, and Yao. "Yima: A Second-Generation Continuous Media Server ", IEEE Computer Magazine, June 2002.
Lesson 11: Security	1. Saltzer, J.H. and Schroeder, M.D., " Protection and the Control of Information in Computer Systems ", Proceedings of the IEEE, 63(9):1278-1308, Sept. 1975. 2. M. Satyanarayanan, " Integrating Security in Large Scale Distributed Systems ", ACM TOCS, Aug. 1989.

Grade Distribution

- **Pre-lab: 2%**
- **Homework (on required background): 3%**
- **Projects**
  - **Project 1: 12% (This project has to be done individually)**
  - **Project 2: 12%**
  - **Project 3: 12%**
  - **Project 4: 12%**

  (Note: Projects 2-4 can be done in groups of 2. It is your own responsibility to find a project partner and work out the logistics of working together. The teaching team will not arbitrate on such matters (e.g., partner dropping the course, etc.). Our assumption is that both partners contribute equally to the project. We will do random sampling of project teams to verify that the project was done with full participation by both members.

  You can choose to do the projects on your own as well without a partner. But there is no special consideration for doing it by yourself as opposed to doing it with a partner.)

- **Class participation**
  - **Ed Discussion Participation: 3%**

  (Provide answers to peer questions; Ask questions; Work out past exams collaboratively, etc.); Note: We will use the summary stats from the forum in the following categories: "views", "contributions", "questions", "answers". The exact weight of each is not revealed to the students, but I am sure the students are savvy enough to know which would count for more!

- **Two paper summaries: 2%**
  Please sign up here: [Paper summary sign up (https://gtvault.sharepoint.com/:x:/s/Fall2024CS6210TAs/EQNce6-aT7hIlo9-k3AFmZIBQcy1tQwJjWZ1eTxmRnUl6Q?e=ZKLwmY)](https://gtvault.sharepoint.com/:x:/s/Fall2024CS6210TAs/EQNce6-aT7hIlo9-k3AFmZIBQcy1tQwJjWZ1eTxmRnUl6Q?e=ZKLwmY)

  (Students sign up on the Wiki and choose two papers from the reading list to write summaries - approximately 1 page for each summary)

- **Tests**
  - **Test 1: 16%**
  - **Test 2: 14%**
  - **Test 3: 12 %**

  The tests will be conducted using Honorlock. You are allowed ONE sheet of BLANK SCRATCH paper at the time of the test for you to do scratch work. You have to show both sides of the paper to the webcam before starting the exam.

  80% of the questions will be released during the exam window. You may collaborate with other students in coming up with answers prior to taking the exam. But the exam itself will be timed, closed notes, taken individually and proctored through Honorlock.

Extra Credit

- **Video Hangout attendance: 0.5% if at least 10 appearances through the semester for the weekly hangout.**
- **We recognize that due to time zone differences it may not be possible for some of you to attend the weekly hangouts at all. We will create an "extra credit assignment" worth 0.5% for students who cannot attend the hangouts on Tuesdays.**
  - **You are to summarize any 10 hangout recordings.**
  - **Each summary should not be more than a page.**
  - **You must aggregate all 10 summaries in a single pdf document and upload it.**
  - **Note: You can eligible for this extra credit option ONLY if you are unable to attend hangouts and mark your attendance through "PointSolutions" during the hangout.**
- **CIOS completion rate at the end of the semester if it exceeds 95% everyone will get 1% added to their course total.**

Hardware Requirements

Summary

- **Students must have access to a machine for code development and a machine for Honorlock exams.**
  - **Bare-metal Linux and Intel-based MacOS are supported for code development, but Linux virtual machines may be used if necessary.**
  - **Only native Windows and MacOS machines are supported for Honorlock exams.**
- **If you need help with hardware support, please ask us via a post in Ed Discussion or office hours.**

- We encourage you to keep your Ed Discussion post public if it may be helpful for other students with similar hardware setup issues.

Requirements for coding assignments

- For code development for assignments, we recommend using a distribution of Linux running on bare metal.
  - If you are using Ubuntu, version 18.04 is recommended.
- If you absolutely must use a hypervisor to bring up a Linux machine, please stick to using VMWare.
  - Georgia Tech provides a student license for this. Students can also use MacOS.
- Note: Project 1 requires support for nested virtualization, so Macs with M series (M1, M2, M3) chips cannot be used.
  - Alternate provisions for these students will be provided when project 1 is released.
- GCC compiler version 7.5.0 will be sufficient for this course. If this changes, we will make an announcement.

The above guidelines are general recommendations for all the projects. Project specific setup requirements will be disclosed when the projects are released.

Requirements for Honorlock

- Exams will be conducted through the use of a software called Honorlock.
- Steps to set up Honorlock and Do's and Don'ts for the exams will be released later.
- Honorlock currently requires access to a native Windows or MacOS machine (not virtualized).
- Do not attempt to use Honorlock with a Windows or MacOS virtual machine within a Linux machine, as this may result in an academic violation as per OSI policy.

Course Policies

All students in this class are expected to know and follow the Georgia Tech Honor Code. You are strongly urged to familiarize yourselves with the GT Student Honor Code (https://osi.gatech.edu/students/honor-code)

Links to an external site. (https://osi.gatech.edu/students/honor-code) rules. Specifically, the following is not allowed:

- Copying, with or without modification, someone else's work when this work is not meant to be publicly accessible *(e.g., a classmate's program or solution).*
- Submission of material that is wholly or substantially identical to that created or published by another person or persons, without adequate credit notations indicating authorship *(plagiarism).*
- Putting your projects on public Github. Otherwise, if a student *(in the future)* copies your codes/projects, the student obviously violates the honor code but you will also be implicated.
- Copying code directly from online resources including ChatGPT is a violation of course policy.

You are encouraged to discuss problems and papers with others as long as this does not involve copying of code or solutions. Any public material that you use *(open-source software, help from a text, or substantial help from a friend, etc...)* should be acknowledged explicitly in anything you submit to us. If

you have any doubt about whether something is legal or not please do check with the class Instructor or the TA.

Collaboration Policy

This class will consist of some individual assignments and some team projects. In order to facilitate a collaborative learning environment, you may discuss the projects and different approaches and techniques, as well as problems you may be having with each other. However, all the code you turn in must be written by you. For written homework, similarly, you may discuss the problems and concepts, but the actual answer that you submit must be written entirely by you.

Please note that any infractions will be reported to the Office of Student Integrity and could lead to suspension from the OMS program.

We will select random subsets of students to interview about their project submissions (since the class is too large to interview everybody about every project). In an interview, you will walk through your code with the TA and explain how it works. Therefore it is important that you understand your own code and are able to explain it.

Late Policy

- **Late submissions are accepted for 24 hours after a deadline with 20% of penalty.**
- **One Time Forgiveness**
  - **Students may choose to avail themselves of the ONE TIME FORGIVENESS policy that allows submission of any of the graded work (except tests) up to ONE WEEK PAST THE POSTED DEADLINE without any penalty. Note that this is ONE TIME ONLY, that is, it cannot be used more than once in the entire semester.**
  - **Students availing this option must create a private Ed Discussion post which will be reviewed by a TA who will grant you the extension on Gradescope.**
  - **OTF is NOT applicable for exams.**

Honor Code

All students in this class are expected to know and follow the Georgia Tech Honor Code. You can read the honor code at this webpage: http://osi.gatech.edu/content/honor-code (Links to an external site.)Links to an external site. (https://osi.gatech.edu/students/honor-code)

Zero tolerance for any infraction. It pains me to say this but there have been incidents of infractions in previous semesters and the consequence is very serious and will affect your academic standing severely. So please do not put yourself (or me) in that situation.