

## UNDERGRADUATE CURRICULUM CHANGE MEMO

**Date:** 11/09/2021

**From:** Dr. Salamah Salamah, Chair of Department Dept. of Computer Science

**Through:** Virgilio Gonzalez, Chair of Curriculum Committee, College of Engineering

**Through:** Louis Everett, Associate Dean for Academic Affairs and Undergraduate Studies, College of Engineering

**Through:** Patricia Nava, Interim Dean, College of Engineering

**To:** Carla Ellis, Chair of University Curriculum Committee

Dr.  
Salamah  
Salamah

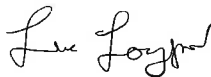
Digitally signed by Dr. Salamah Salamah  
DN: cn=Dr. Salamah Salamah, o=UTEP, ou=Computer Science, email=salamah@utep.edu, c=US  
Date: 2021.11.09 14:00:48 -0700

**Proposal Title:** Computer Science curriculum changes for the Fall 2022 catalog

---

The Department of Computer Science's Systems Area Committee reviewed our Systems courses and proposed changes to the title and description of CS 3432 and 4375 courses. The proposal was presented and approved by the Computer Science faculty.

**Rationale.** To reflect the current content and goals of our two required systems course, it is necessary to modify the title of CS 3432 from "Computer Architecture I: Comp Org/Design" to "Computer Organization" and modify the title of CS 4375 from "Theory of Operating Systems" to "Operating Systems Concepts". In addition, we need to update the course descriptions of the two courses.



Digitally signed by Luc Longpré  
DN: cn=Luc Longpré, o=UTEP,  
ou=Department of Computer  
Science, email=lolongpre@utep.edu,  
c=US  
Date: 2021.11.09 13:41:23 -07'00'

# CURRICULUM CHANGE PROPOSAL

## APPROVAL PAGE

**Proposal Title: Computer Science curriculum changes for the Fall 2022 catalog**

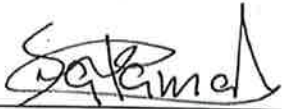
**College: Engineering**

**Department: Computer Science**

### DEPARTMENT CHAIR

---

**I have read the enclosed proposal and approve this proposal on behalf of the department.**



11-09-2021

**Signature**

**Date**

### COLLEGE CURRICULUM COMMITTEE CHAIR

---

**I have read the enclosed documents and approve the proposal on behalf of the college curriculum committee.**

**Signature**

**Date**

### COLLEGE DEAN

---

**I have read the enclosed documents and approve the proposal on behalf of the college. I certify that the necessary funds will be allocated by the college in support of this proposal.**

**Signature**

**Date**

**From:** [Granda, Virginia D](#)  
**To:** [Rivera, Julie A](#)  
**Subject:** FW: COECC UG Proposal November 12, 2021  
**Date:** Friday, November 12, 2021 4:51:32 PM  
**Attachments:** [image014.png](#)  
[image015.png](#)  
[image016.png](#)  
[image003.png](#)  
[image004.png](#)  
[image005.png](#)  
[image017.png](#)  
[image018.png](#)  
[CS-Computer-Science-Curriculum-Changes--Fall-2022-Catalog.pdf](#)  
[image006.png](#)

---

Julie,

Attached is the UG proposal that was approved by our college.

Please see below for the approvals.

Have a great weekend,

Virginia



**Virginia Granda-Becker**

Coordinator for Undergraduate Studies and Academic Affairs

College of Engineering  
The University of Texas at El Paso  
500 W. University Ave  
El Paso, TX 79968  
Office: (915) 747-8011  
[www.utep.edu/engineering/eec](http://www.utep.edu/engineering/eec)

---

**From:** [Nava, Patricia A.](#)  
**Sent:** Friday, November 12, 2021 4:22 PM  
**To:** Granda, Virginia D <[granda@utep.edu](mailto:granda@utep.edu)>  
**Subject:** RE: COECC UG Proposal November 12, 2021

Yes, this is approved.



**Patricia A. Nava, Ph.D.**

Interim Dean  
Professor of Electrical and Computer Engineering  
El Paso Electric Professor in Education  
UTEP Distinguished Teaching Professor

College of Engineering  
The University of Texas at El Paso  
500 West University Avenue  
El Paso, TX 79968-0521

Office: 915-747-6917  
Fax: 915-747-5437  
[utep.edu/engineering](http://utep.edu/engineering)



---

**From:** Granda, Virginia D  
**Sent:** Friday, November 12, 2021 4:06 PM  
**To:** Nava, Patricia A. <[pnav@utep.edu](mailto:pnav@utep.edu)>  
**Subject:** FW: COECC UG Proposal November 12, 2021

Good afternoon Dr. Nava,

Attached is the UG proposal that was approved by our COECC, its chair and the Associate Dean of UG Studies and Academic Affairs.

Please reply letting me know if you approve it.

Best Regards,

Virginia



**Virginia Granda-Becker**  
Coordinator for Undergraduate Studies and Academic Affairs  
  
College of Engineering  
The University of Texas at El Paso  
500 W. University Ave  
El Paso, TX 79968  
Office: (915) 747-8011  
[www.utep.edu/engineering/eec](http://www.utep.edu/engineering/eec)

---

**From:** Everett, Louis  
**Sent:** Friday, November 12, 2021 4:00 PM  
**To:** Granda, Virginia D <[granda@utep.edu](mailto:granda@utep.edu)>  
**Subject:** RE: COECC UG Proposal November 12, 2021

I approve this.

---

**From:** Granda, Virginia D  
**Sent:** Friday, November 12, 2021 4:00 PM  
**To:** Everett, Louis <[leverett@utep.edu](mailto:leverett@utep.edu)>

**Subject:** FW: COECC UG Proposal November 12, 2021

Good afternoon Dr. Everett,

Attached is the UG proposal that was approved by our COECC and its chair.

Please reply letting me know if you approve it.

Best Regards,

Virginia



**Virginia Granda-Becker**  
Coordinator for Undergraduate Studies and Academic Affairs  
  
College of Engineering  
The University of Texas at El Paso  
500 W. University Ave  
El Paso, TX 79968  
Office: (915) 747-8011  
[www.utep.edu/engineering/eec](http://www.utep.edu/engineering/eec)

---

**From:** Gonzalez, Virgilio  
**Sent:** Friday, November 12, 2021 3:58 PM  
**To:** Granda, Virginia D <[granda@utep.edu](mailto:granda@utep.edu)>  
**Subject:** RE: COECC UG Proposal November 12, 2021

I approve it.



**Virgilio Gonzalez, Ph.D.**  
Associate Chair  
E-mail: [vgonzalez3@utep.edu](mailto:vgonzalez3@utep.edu)  
  
Electrical and Computer Engineering Department  
The University of Texas at El Paso  
500 W. University Ave. , ENGR-A325  
El Paso, TX 79968  
Office: 915-747-6622  
Fax: 915-747-7871  
[ece.utep.edu](http://ece.utep.edu)



---

**From:** Granda, Virginia D <[granda@utep.edu](mailto:granda@utep.edu)>  
**Sent:** Friday, November 12, 2021 15:50

**To:** Gonzalez, Virgilio <[vgonzalez3@utep.edu](mailto:vgonzalez3@utep.edu)>  
**Subject:** COECC UG Proposal November 12, 2021

Good afternoon Dr. Gonzalez,

Attached is the UG proposal that was approved by our COECC today.

Can you please reply letting me know if you approve it as the COECC chair?

Best Regards,

Virginia



**Virginia Granda-Becker**  
Coordinator for Undergraduate Studies and Academic Affairs  
  
College of Engineering  
The University of Texas at El Paso  
500 W. University Ave  
El Paso, TX 79968  
Office: (915) 747-8011  
[www.utep.edu/engineering/eec](http://www.utep.edu/engineering/eec)

# COURSE CHANGE FORM

All fields below are required

---

College : ENGINEERING

Department : COMPUTER SCIENCE

Rationale for changing the course:

CS3432 is not a computer architecture course, but rather a first systems course in computer organization with a focus on the hardware-software interface. Thus, it is proposed to change the course title to Computer Organization.

All fields below are required

---

Subject Prefix and number CS 3432

Course Title Comp. Arch. I: Computer Organization and Design

Change	From	To
Ex. Prerequisite	Ex. POLS 2310	Ex. POLS 2312
Course Title	Computer Architecture I: Computer Organization and Design	Computer Organization

These changes will be reflected in Banner, Goldmine, and the catalog

# COURSE CHANGE FORM

All fields below are required

---

College : ENGINEERING

Department : COMPUTER SCIENCE

Rationale for changing the course:

Focusing on what students most need to know, for success in the workplace and for advanced system courses, and to better match what the accreditors now want, we are updating the course coverage and accordingly also the course description. The title is also being changed to more accurately describe the focus and coverage.

All fields below are required

---

Subject Prefix and number CS 4375

Course Title ... Operating Systems ...

Change	From	To
Ex. Prerequisite	Ex. POLS 2310	Ex. POLS 2312
Course Description	Process and thread management, concurrency, memory management, processor scheduling, I/O management and disk scheduling, and file management	Process and thread management, processor scheduling and concurrency, interprocess communication, memory management, input/output management, file systems, and networking basics
Course Title	Theory of Operating Systems	Operating Systems Concepts

These changes will be reflected in Banner, Goldmine, and the catalog



## CS 3432 Computer Organization and Design (Arch 1)

Fall 2021 Syllabus

Lecture: CRN 17591; TR 9:00-10:20am; Geology 123

Lab: CRN 17602; TR 10:30-11:50am; CCSB 1.0704

Instructor: Shirley Moore

Course webpage: <http://svmoore.pbworks.com/>

CS 3432 is a first systems course about how computer systems work. Whereas computer architecture describes the functional behavior of a computer, computer organization describes the structural relationships. CS 3432 is a course in computer organization that answers questions such as the following:

1. How are programs written in a high-level language, such as C or Java, translated into the language of the hardware, and how does the hardware execute the resulting program?
2. What is the interface between the software and the hardware, and how does software instruct the hardware to perform the necessary functions?
3. What determines the performance of a program, and how can hardware designers or programmers improve the performance?
4. What are the reasons for and the consequences of the recent switch from sequential to parallel processing?

### ▪ Teaching Team

- Instructor: Shirley Moore, [svmoore@utep.edu](mailto:svmoore@utep.edu)
  - Office: CCSB 3.0608
  - Office phone: 915-747-5054
  - Office hours: T 1:30-2:30pm, W 12:00-1:00pm, others by appointment (in person or virtual)
  - Webpage: <http://www.cs.utep.edu/svmoore>
- Teaching Assistant: Steven Ibarra, [sibarra7@miners.utep.edu](mailto:sibarra7@miners.utep.edu)
- Instructional Assistants: Esteban Munoz, [emunoz22@miners.utep.edu](mailto:emunoz22@miners.utep.edu)  
Alan Perez, [aeperez8@miners.utep.edu](mailto:aeperez8@miners.utep.edu)

### ▪ Prerequisites

- To succeed in this course, you need familiarity and maturity with concepts and techniques taught in digital design (EE 2369/2169), elementary data structures and algorithms (CS 2401), and discrete math (Math 2300).
- As stipulated in the course catalog, the usual way to demonstrate this familiarity is by earning a C or above in (1) all of these courses and (2) Data Structures (CS 2302).
- Students who earn B's or above in EE 2369/2169, CS 2401, and Math 2300 and are taking CS 2302 concurrently, are also considered ready.

### ▪ Lab Section

- Students **must** enroll in the associated lab section.
- Lab sessions and assignments will be managed by the TA.
- **Participation is mandatory.** It is extremely easy to fall behind and imperative that you make arrangements with the instructor or TA to make-up missed lessons and work. Students at risk of failing due to not engaging with the lab section may be dropped.
- There will be scheduled and unscheduled quizzes in the lab section.

- **Texts and Readings**
  - Required: David A. Patterson and John L. Hennessy. Computer Organization and Design RISC-V Edition: The Hardware Software Interface, Second edition, Morgan Kaufmann, ISBN: 978-0128203316.
  - Required: Kernighan, Brian W & Ritchie, Dennis M. The C Programming Language, Second edition, Prentice Hall, ISBN: 0-13-115817-1.
  - Other readings and resources will be posted on the course webpage.
- **Learning Outcomes**
  - Learning outcomes are in the appendix at the end of this document.
- **Computers**
  - You will need access to a computer capable of running Linux (either natively or using virtualization software) to participate in class activities. You will need to install an ANSI C compiler prior to the start of lab sessions the second week of class. Should you not have access to an appropriate machine, the department has a limited number of portable computers that can be lent out to students.
- **Attendance**
  - Attendance is mandatory at all lecture and lab sessions unless special circumstances are arranged ahead of time with the instructor, or as soon afterwards as possible in the event of an emergency. Lecture sessions will be recorded, but viewing the recorded session is not a substitute for attending class. Breakout groups and hands-on exercises will be part of lecture sessions and it will not be possible to records those portions.
- **Grading Breakdown**

1000-900 = A    899-800 = B    799-700 = C    699-600 = D    599 and Below = F

  - Lab assignments and quizzes: 40% (400 points)
  - Class participation and homework: 20% (200 points)
  - Midterm exam: 20% (200 points)
  - Final exam: 20% (200 points)
- **Course Schedule**
  - The course schedule of topics and assignments will be posted on the course website.

### Homework, Tests, and Labs

- **Homework**
  - Students are expected to review topics taught in class, work on solutions to assigned problems, and be able to demonstrate skills and solutions during class. Homework assignments will be posted on Teams with instructions on how to submit solutions.
- **Exams and quizzes**
  - Quizzes
    - Quizzes assess students' abilities to demonstrate knowledge, to design solutions to realistic problems, and to present these solutions in a clear and professional fashion.
    - Quizzes can cover any concept or skill previously taught in the course, are generally offered at the rate of one per week and are generally unannounced (so students must be continuously prepared).

- Midterm and Final Exams
  - The midterm exam will occur approximately midway through the course.
  - The final exam date is scheduled by the university based on lecture time.
  - Like quizzes, exams may cover any concept or skill previously taught in the course.
- The following are prohibited during quizzes and exams unless unambiguous and explicit permission is provided by the instructor:
  - Collaboration or communication with others.
  - Looking up answers in books, on the Internet, etc.
- **Labs**
  - Labs are intended to provide an opportunity for students to explore and practice the use of tools and to apply concepts presented in class within the context of programming projects.
  - Lab assignments will be posted on Teams.
  - Lab assignments will be graded during in-person demonstrations with individual students.
    - Demonstrations must occur within one week following the lab due date.
    - Demonstrations may require students to modify their programs and demonstrate competency with development tools.
  - Students are expected to act professionally
    - By reading and studying relevant resources
    - By attributing credit to any person or reference materials that substantively contributed to their solutions
    - By only submitting solutions they fully understand
    - Professionalism includes honesty, clarity, and accuracy.
    - Students are encouraged to help each other develop and tune their lab projects. There is no penalty for collaboration as long as it is documented.
  - Requirements
    - **Functional:** Assignments will require students to create complete programs or designs or modify programs or designs provided by the instructor.
    - **Documentation:** Submissions should include documentation that facilitates the grader's determination of
      - How to compile, use, and test
      - Principles of operation (e.g., comments and other descriptive prose)
      - Elements of the submission that were developed by others. This includes both algorithms and code. Vague attributions of credit (e.g., "Assistance was received from Jim Smith.") are unacceptable.
    - **Completeness:** Students whose labs do not substantially satisfy functional and documentation requirements will receive no credit for that lab.

### Disabilities and Accommodations

- If you have a disability and accommodations, please contact The Center for Accommodations and Support Services (CASS) at 747-5148, or by email to [cass@utep.edu](mailto:cass@utep.edu). For additional information, please visit the CASS website at [www.sa.utep.edu/cass](http://www.sa.utep.edu/cass).

## **Academic Honesty**

- Students are expected to conduct themselves in a professional and courteous manner, as prescribed by the Standards of Conduct: <https://www.utep.edu/hoop/section-2/student-conduct-and-discipline.html>
- Submitted work should be unmistakably your own. You may not transcribe or copy a solution taken from another person, book, or other source (e.g., a web page). Copying other's work will not be tolerated. Professors are required to report academic dishonesty and any other violation of the Standards of Conduct to the Dean of Students.
- Permitted collaboration: Students may discuss requirements, background information, test sets, solution strategies, and the output of their programs. However, implementations and documentation must be their own creative work. Students are required to document advice received from others and all resources utilized in the preparation of their assignments.
- If academic dishonesty is suspected: The Dean of Students office will be contacted for adjudication. A temporary "incomplete" grade will be issued if their investigation extends beyond the grading period.

## **COVID-19 Precaution Statement**

- Please stay home if you have been diagnosed with COVID-19 or are experiencing COVID-19 symptoms. If you are feeling unwell, please let me know as soon as possible, so that we can work on appropriate accommodations. If you have tested positive for COVID-19, you are encouraged to report your results to [covidaction@utep.edu](mailto:covidaction@utep.edu), so that the Dean of Students Office can provide you with support and help with communication with your professors. The Student Health Center is equipped to provide COVID-19 testing.
- The Center for Disease Control and Prevention recommends that people in areas of substantial or high COVID-19 transmission wear face masks when indoors in groups of people. The best way that Miners can take care of Miners is to get the vaccine. If you still need the vaccine, it is widely available in the El Paso area, and will be available at no charge on campus during the first week of classes. For more information about the current rates, testing, and vaccinations, please visit [epstrong.org](http://epstrong.org).

## Appendix. CS 3432 Learning Outcomes

### Level 1

- A1. Define and explain the purpose of an instruction set architecture (ISA).
- HSI2. Explain the relationship and differences between a high-level programming language, assembly language, and machine language.
- HSI3. Describe the fetch-execute cycle in terms of the hardware-software interface between machine instructions and processor components.
- NR1. Explain the relationship between a high-level language basic data type (e.g., signed or unsigned integer, floating point number) and its representation as a bit pattern inside the computer.
- A2. Describe the basic components of a processor (e.g., register file, special-purpose registers, control unit, memory) and how they interact with one another.
- HSI4. Explain how procedures are supported by processor hardware.
- HSI5. Explain exception/interrupt handling in terms of the hardware-software interface.
- HSI6. Explain various ways an operand can be addressed in an assembly language instruction.
- HSI7. Describe the process of compiling/assembling, linking, loading, and executing a program.

### Level 2

- NR2. Convert between different integer data representations (e.g., decimal, binary, hexadecimal, octal).
- NR3. Interpret the bit representation of a floating-point number.
- NR4. Perform addition and subtraction on two's complement representation of integers.
- NR5. Use bitwise operators to access and manipulate values stored in a subset of bits within a byte or word.
- NR6. Determine range and precision (if applicable) of numbers that can be stored for a given data type and determine whether an integer operation will result in overflow.
- HSI8. Convert between machine and assembly language representations of instructions – i.e., encode and decode instructions.
- HSI9. Trace the datapath through the processor for a given class of instructions (e.g., arithmetic-logical, memory access, conditional branch)
- HSI10. Trace the execution of an assembly language program with procedure calls in terms of allocation and deallocation of stack frames.
- L1. Translate expressions and assignment statements from C to assembly language.
- L2. Translate Boolean logic and control flow constructs (decisions, loops) from C to assembly language.
- L3. Translate operations on arrays, structs, and pointers from C to assembly language.

### Level 3

- HSI11. Implement/debug simple imperative programs in assembly/machine language.
- HSI12. Write or call a procedure with local variables, parameters, and return value in assembly language.
- HSI13. Implement a simple interrupt handler.
- T1. Compose, compile/assemble, execute, and debug simple programs in a command-line environment, using appropriate modularization and multiple files.

# CS 4375: Theory of Operating Systems

Fall 2021

**Instructor:** Nigel Ward, nigel@utep.edu  
Office Hours: T 2:30 - 3:30 and W 3:50 – 4:50, or by appointment,  
Computer Science 3.0408, also 747-6827

**TA:** Jonathan E. Avila, jeavila6@miners.utep.edu  
Office Hours: Monday 12:30-1:30pm CCSB G.0512 ; Wednesday 1:30-2:30pm  
CCSB 1.0706 ; Friday 2:00-4:00pm CCSB 1.0706 ; or by appointment

**Class Time:** MW 10:30 – 11:50, Education Building, room 302

**Textbooks:** *Operating Systems: Three Easy Pieces*, by Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau.

*Computer Networks: A Systems Approach*, by Larry Peterson and Bruce Davie.

**Course Description:** Process and thread management, processor scheduling and concurrency, interprocess communication, memory management, input/output management, file systems, and networking basics

**Goal:** Introduce concepts that will be foundational for further study, whether academic or professional, of 1) Computer Security and Forensics, 2) Systems Administration and Network Administration, and 3) Systems Programming and Network Programming, including developing for embedded systems, cloud systems, and high-performance systems.

## Course Policies

The prerequisite for this class is CS 3432 with a C or better.

Assigned readings are to be done before class.

Bring to class a whiteboard marker or two and the relevant textbook, either hardcopy or on your device, or on your buddy's device.

Assignments are to be submitted in hardcopy, unless otherwise specified. The maximum points attainable for late assignments will be reduced by 10% per day or partial day of lateness, for up to five days, or more if the lateness impedes participation in class or the assignment is received after the solution has been discussed. Email submissions of assignments are not accepted unless otherwise specified.

Assignments are to be done individually unless specifically designated as group assignments. While you may discuss assignments with others, your solutions should be designed, written/assembled, and tested by you alone. If you need help, consult the TA or the instructor.

The use of found code and shared code is acceptable, unless otherwise specified, provided that you acknowledge your sources, state specifically what you used, and understand every line of code.

Programming assignments will be graded primarily on functionality, design quality, thoroughness of testing, and readability. Some of these factors inevitably involve subjective judgments; if you have questions about the criteria, please see the TA or the instructor.

Tests will be closed-book, except that one single-sided page of hand-written notes may be used for the first test, two for the second test, and three for the final. If you leave the room for any reason, your test will be graded on only what you did up until that time. No make-up exams or assignments will be

given except under the conditions set forth in the Catalog.

Grades will be based on four components, weighted approximately as follows: 45% assignments, 22% final examination, 28% tests, and 5% other factors, including quizzes, in-class exercises, and participation.

Assignments and tests will be challenging. Grading will be on a points-earned basis (points above zero), rather than a points-off basis (points below expectation). Letter grades will be assigned accordingly: the A/B break will probably be around 80% and the B/C break around 70%. Final grades may be adjusted upwards in cases where performance on both the assignments and the tests is solid.

Students are free to attend class or not, bearing in mind that absence may annoy other students, interfere with learning, and result in a lower grade.

### General Policies

No make-up exams or assignments will be given except under the conditions set forth in the Catalog.

Students are expected to be punctual, and, as always, to conduct themselves professionally and courteously.

If you have or suspect a disability and need accommodation, contact the Center for Accommodations and Support Services at 747-5148 or at [cass@utep.edu](mailto:cass@utep.edu) or visit Room 106 Union East.

### Topics, Readings and Major Assignments, tentative

Introduction	(3 days)	<i>Python warm-up assignment</i>	ADAD Ch. 1-4
Processes	(3 days)	<i>Shell assignment</i>	ADAD Ch 5-7, 11
Memory	(3 days)	<i>sundry exercises</i>	ADAD Ch 12-18
			Test 1
Sockets	(3 days)	<i>Sockets assignments</i>	various readings
Threads and Concurrency	(5 days)	<i>various assignments</i>	ADAD 25 - 26; 28 - 32, portions
			Test 2
Networking	(5 days)	<i>sundry exercises</i>	ADAD 48; PD Ch 1, §2.1, §3.1 thru 3.1.1, §3.3 thru 3.3.5, Ch 5 thru 5.2.3, §7.1, Ch 8 thru 8.2, §8.5.6, §9.3.1
Files and Devices	(4 days)	<i>sundry exercises</i>	ADAD Ch. 35-37, 39-40; 38-46 portions
			Final Exam

**Course Website:** <http://www.cs.utep.edu/nigel/os/>

### Important Dates (tentative)

August 23: Class begins

September 6: Labor Day

September 27: Test 1  
November 1: Test 2  
December 3: Dead Day  
December 10: Final Exam, 10:00-12:45

### **Level 1 Outcomes: Knowledge and Comprehension**

Level 1 outcomes are those in which the student has been exposed to the terms and concepts at a basic level and can supply basic definitions. The material has been presented only at a superficial level.

Upon successful completion of this course, students will be able to:

- V1i. Choose a scheduling approach suitable for given simple problem.
- V1j. Explain segmentation and its security implications.
- V1l. Explain some ways in which virtualization creates vulnerabilities.
- V1m. Describe and motivate the components of process and virtual machine context..
- V1n. Explain the need for paging and the basics of demand loading.
- V1o. Describe the motivation for and gross characteristics of a trusted computing base.
- V1x. Explain how domain names, IP addresses, and file names work and why.
- C1c. Given an application, identify the factors relevant to choosing a synchronous or asynchronous solution.
- E1f. Choose when to use datagram versus virtual-circuit communication.
- E1h. Differentiate transmission and propagation latencies and some factors affecting them.
- E1i. Explain how data is serialized (byte order, representation, buffering).
- E1j. Explain the difference between lossy and lossless compression.
- E1l. Interpret the output of a packet capture tool.
- E1n. Explain the role of cryptographic hashes and symmetric and assymetric keys in security.
- E1o. Explain the functionality handled at the physical, link, network, and transport layers.
- E1p. Explain the functionality handled at different network layers.
- E1q. Explain some data structures used for storing files on disk.
- E1r. Explain the memory hierarchy, locality, and redundancy.
- E1s. Explain generic device APIs, including the bidirectional handling of interrupts and requests.

### **Level 2 Outcomes: Application and Analysis**

Level 2 outcomes are those in which the student can apply the material in familiar situations, e.g., can work a problem of familiar structure with minor changes in the details.

Upon successful completion of this course, students will be able to:

- V2q. Use the concepts of process state and state transition to characterize system and process behavior.
- V2r. Relate the distinction between supervisor and user permissions to the design and implementation of system calls.
- V2t. Write programs that use interprocess communication, specifically pipes and/or sockets.
- V2u. Use simple system calls for common needs.
- C2g. Implement producer-consumer coordination.
- C2h. Build a server-side program that uses multi-threading to handle multiple simultaneous clients.
- C2i. Identify situations where deadlock may occur, and suggest ways to prevent it.
- A2g. Perform simple arithmetic computations related to major families (e.g. determine page number or whether an address is within a power-of-2 segment)

### **Level 3 Outcomes: Synthesis and Evaluation**

Level 3 outcomes are those in which the student can apply the material in new situations.

This is the highest level of mastery.

Upon successful completion of this course, students will be able to:

- V3p. Choose among virtual machines, processes, containers and sandboxes as ways to support common programmer needs.
- V3w. When a process or a computer is running too slowly, infer some probable causes.
- C3j. Distinguish when blocking vs nonblocking calls are appropriate.