**This is an example of a Breadth Requirement Petition Packet and should be used solely for example purposes only **

UCLA ENGINEERING
Computer Science

BREADTH REQUIREMENT

This is an Interactive PDF – please type responses.

Student: Bruin (Last Name) Josie (First Name) Degree Objective: M.S.

Email: josiebruin@g.ucla.edu Phone Number: (xxx) xxx-xxxx UID: XXXXXXXXX

Date: month/day/year Advisor:

Please fill in all fields above - Advisor field can be blank

Only Grades of B- or Higher May Be Used

If your institution has a numeric system, enter marks over total possible (eg - 90/150)

Course	Option 1 UCLA Course	Option 2: Equivalent Course Attach a copy of your transcript, and hilite courses	Option 3 Credit by Exam
	Quarter: Grade:	School: Course Title: Quarter and Year: Final Grade: Row:	Signature: Date: ***See list below***
**Please be sure to select the UCLA equivalent course in the Course column as seen below (I selected CS 180) **	Quarter: Grade:	School: Course Title: Quarter and Year: Final Grade: Row:	Signature: Date: ***See list below***
	Quarter: Grade:	School: Course Title: Quarter and Year: Final Grade: Row:	Signature: Date: ***See list below***
	Quarter: Grade:	School: Course Title: Quarter and Year: Final Grade: Row:	Signature: Date: ***See list below***
CS 180 - Theory *Make sure to fill in all the fields minus the row number in the Option 2 Column. If the course you are petitioning is listed on the equivalent list, no need to submit a petition packet.*	Quarter: Grade:	School: University of Illinois at Urbana-Champaign Course Title: CS/ECE 374 Intro to Algs & Models of Cc Quarter and Year: Fall 2018 Final Grade: B+ Row:	Signature: Date: ***See list below***

CS 201 Seminars

	First	Second	Third
Quarter			
Grade			

Credit by Exam Certifying Instructors

CS 111: Eggert
CS 118: Lu or Zhang
CS 151B: Reinman or Tamir
CS 130: Eggert
CS 131: Millstein
CS 132: Palsberg
CS 143: Cho or Zaniolo
CS 161: Darwiche or Korf
CS 174A: Terzopolous
CS 180: Ostrovsky

For office use only:

( ) **APPROVED** _____ _____
Grad Affairs Office Staff date

Updated July 2019

****This is solely for example purposes only ****

CS/ECE 374: About This Course

CS/ECE 374 covers fundamental tools and techniques from theoretical computer science, including design and analysis of algorithms, formal languages and automata, computability, and complexity. Specific topics include regular and context-free languages, finite-state automata, recursive algorithms (including divide and conquer, backtracking, dynamic programming, and greedy algorithms), fundamental graph algorithms (including depth- and breadth-first search, topological sorting, minimum spanning trees, and shortest paths), undecidability, and NP-completeness. The course also has a strong focus on clear technical communication.

Prerequisites

We assume that students have *mastered* the material taught in CS 173 (discrete mathematics, especially induction) and CS 225 (basic algorithms and data structures). Note that "mastery" is not the same as "exposure" or even "a good grade"; hence, Homework Zero! **If you got a C+ or worse in CS 173, we strongly recommend retaking 173 before taking 374.**

The CS department is now **strictly** enforcing the CS 173 and CS 225 prerequisites. Students must have credit for both courses (either through taking the class, passing the proficiency exam, or transferring credit from an equivalent course taken elsewhere) before they can register for 374.

Requirements

This course is required for all undergraduates majoring in Computer Engineering or any species of Computer Science.

Postrequisites

CS/ECE 374 is a formal prerequisite for at least the following classes:

- CS 421: Programming Languages
- CS 473: Algorithms

Coursework

Course grades are based on weekly written homeworks (30% total), two midterms (20% each), and a final exam (30%). See the grading policies for more details.

Difficulty

Many students consider 374 to be the most challenging course in the entire undergraduate CS/CE curriculum (perhaps after ECE 391). On the other hand, we believe (and employers and alumni seem to agree) that 374 is also the most *useful* course in the undergraduate CS/CE curriculum (perhaps after CS 225), in no small part *because* it is so challenging. CS and CE majors are among the brightest and hardest-working students on campus; an easier course would be an insulting waste of your time.

❖ Class Resources ❖

Web site

Almost everything—course policies, detailed schedule, lecture notes, lecture videos, homeworks, homework solutions, lab problems, etc.—can be found here. Hey, look! You found it!

Reading

There is no required textbook. Lecture notes / book chapters for all lecture topics are available on the schedule page; Jeff will revise and update these as the semester progresses.

The algorithms portion of the class largely follows Jeff's *Algorithms* textbook, which is freely available online. The book web site also contains lots of additional lecture notes, along with homeworks, lab handouts, and exams (but no solutions) from his past theory classes.

You may also find resources from other Illinois instructors useful:

- Fall 2015 offering of CS 374 by Chandra Chekuri and Manoj Prabhakaran
- Margaret Fleck's discrete mathematics textbook for CS 173 (review material)
- Mahesh Viswanathan's automata and formal languages notes for CS 373 (Spring 2013)
- Margaret Fleck and Sariel Har-Peled's automata and formal languages notes for CS 373 (Spring 2008 and Spring 2009)

Videos

Recordings of all lectures automatically appear on two separate sites (Echo360 and MediaSpace) at most a few hours after each lecture. Streamable and downloadable lecture videos are also available on the schedule page at most a day after each lecture. However, we strongly encourage students to attend the lectures in person to get the most out of them.

Both the Echo360 videos and the downloadable videos should be freely available to the public forever; the MediaSpace videos require university credentials to access.

Jeff's videos from Spring 2018 are also publicly available.

Gradescope

We will use Gradescope for homework submission and grading (both homeworks and exams). Anyone can sign up for access to the CS/ECE 374 Gradescope site with any name and any email address, using the self-enrollment code **MKNYRY**. We will separately ask you for your Gradescope identity, so that we can map your homework grades to you.

Piazza

We will use Piazza for online discussions. Anyone can sign up for access to the CS/ECE 374 Piazza site with with any name and any email address; no access code is required. We strongly encourage posting questions on any course-related topic to Piazza rather than emailing the course staff. You can even post your questions anonymously. (However, we can only give you extra credit for helpful posts if you post them using your real name.)

Etc.

We've collected a long list of other useful resources on a separate page.

****This is solely for example purposes only****

CS/ECE 374: Lecture and Lab Schedule

The calendar below lists the topics of each lecture and lab section for the semester, with links to relevant lecture notes, slides, lecture videos, and lab handouts. **Topics for future lectures and labs are subject to change; exam dates are not.** Lecture videos are available at Echo360 site and requires university credentials to login.

Week	Tuesday Lecture	Wednesday Lab	Thursday Lecture	Friday Lab
Aug 28-31	Administrivia and course goals Notes on strings [sect A: slides1, slides2] [sect B: slides1, slides2]	String induction [Jeff's induction notes, Chandra's additional notes] [solutions]	Languages and regular expressions [sec A slides, sect B slides]	Regular expressions [solutions]
Sep 4-7	DFAs, product construction [Automata Tutor] [sec A slides] [sec B slides] DFA Proofs	DFA construction [solutions]	Non-determinism, NFAs Jeff's NFA notes [sec A slides] [sec B slides]	DFA product construction [solutions]
Sep 11-14	NFAs continued, equivalence with DFAs and regular expressions Jeff's NFA notes [sec A slides sec B slides]	DFA/NFA transformation [solutions]	Proving non-regularity via fooling sets [slides] [DFA Proofs][Fooling sets guide]	Proving nonregularity [solutions]
Sep 18-21	Context-free languages and grammars [slides: secA, secB]	Context-free grammars [solutions]	Turing machines: history, formal definitions, examples, variations [slides]	Regular or not? [solutions]
Sep 25-28	Universal Turing Machine, RAM simulation, P [UTM slides][complexity slides]	Turing machine design [solutions]	**SecA** *Optional review for Midterm 1* **Sec B** Recursion: Hanoi, mergesort (see Oct 2 links) sec B slides	*Optional review for Midterm 1*
Midterm 1: Monday, October 1, 7-9:30pm				
Oct 2-5	**Sec A:** Recursion: Hanoi, mergesort **Sec B**: no lecture [slides][solving recurrences]	Hint: Binary search [solutions]	Divide and conquer: linear-time selection, Karatsuba multiplication [recurrence notes] [slides]	Divide and conquer [solutions]
Oct 9-12	Backtracking: independent set, longest increasing subsequence [sec A slides sec B notebook]	Backtracking [solutions]	Dynamic programming: splitting strings, longest increasing subsequence [slides sec B notebook]	Dynamic programming [solutions]
Oct 16-19	More DP: Edit Distance, MIS in trees [slides]	More dynamic programming [solutions]	CYK Algorithm, Graphs, basic search [CYK slides, Graph search slides]	Yet even still more dynamic programming **Drop deadline** [solutions]
Oct 23-26	Directed graphs, depth-first search [slides]	Graph modeling [solutions]	Catch up lecture [Graph notes]	Graph modeling [solutions]
Oct 30 - Nov 2	BFS and Shortest Paths [slides]	Shortest paths [solutions]	Shortest Paths with Negative Lengths via DP [slides]	More shortest paths [solutions]
Nov 6-9	MST Algorithms [Kent's slides], [slides]	MST [solutions]	Optional review for midterm 2 [sec B sketches]	*Optional review for Midterm 2*
Midterm 2: Monday, November 12, 7-9:30pm **Conflict exam: Tuesday, Nove 13**				
Nov 13-16	Greedy algorithms [slides]	Greedy [solutions]	Undecidability: halting problem, diagonalization, reductions [slides]	Undecidability via reductions [solutions]
Nov 20-23	**Fall break**			

Week	Tuesday Lecture	Wednesday Lab	Thursday Lecture	Friday Lab
Nov 27-30	Polynomial time Reductions [slides]	Self Reductions [solutions]	NP, NP-Completeness, 3SAT to Independent Set [slides][helpful video on P vs NP]	NP-hardness proofs [solutions]
Dec 4-7	More NP-hardness reductions: 3-coloring, Hamiltonian cycle [slides]	NP-Hardness, the final chapter [solutions]	Undecidability: more reductions, Rice's theorem [slides] **ICES Forms**	Using Rice's Theorem **ICES Forms** [solutions]
Dec 11-14	**No lecture, review session Thursday**	Review for final	Review session: 7pm, ECEB 1002	
Final exam: Saturday, Dec 15, 1:30-4:30 p.m.				

This is solely for example purposes only

UNIVERSITY OF ILLINOIS AT URBANA - CHAMPAIGN

Urbana, Illinois 61801

Student Name

University

Issue Date:

Level: U

Day - Month of Birth:

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:						
	Ehrs: 4.00	GPA-Hrs: 4.00	QPts: 16.00	GPA: 4.00		
Summer 2018 - Urbana-Champaign						
Grainger Engineering						
Computer Science						
PHYS	214	Univ Physics: Quantum Physics	2.00	A	8.00	
	Ehrs: 2.00	GPA-Hrs: 2.00	QPts: 8.00	GPA: 4.00		
Fall 2018 - Urbana-Champaign						
Grainger Engineering						
Computer Science						
CS	374	Intro to Algs & Models of Comp	4.00	B+	13.32	
CS	397	Individual Study	3.00	A	12.00	
CS	498	Applied Cryptography	3.00	A-	11.01	
PHYS	225	Relativity & Math Applications	2.00	A	8.00	
THEA	101	Introduction to Theatre Arts	3.00	A	12.00	
	Ehrs: 15.00	GPA-Hrs: 15.00	QPts: 56.33	GPA: 3.75		
Spring 2019 - Urbana-Champaign						
Grainger Engineering						
Computer Science						
ASTR	100	Introduction to Astronomy	3.00	A+	12.00	
CS	210	Ethical & Professional Issues	2.00	A	8.00	
CS	499	Senior Thesis	3.00	A	12.00	
MATH	413	Intro to Combinatorics	3.00	A-	11.01	
MATH	461	Probability Theory	3.00	A	12.00	
	Ehrs: 14.00	GPA-Hrs: 14.00	QPts: 55.01	GPA: 3.92		
Deans List						
Fall 2019 - Urbana-Champaign						
Grainger Engineering						
Computer Science						
CS	357	Numerical Methods I	3.00	A	12.00	
DANC	100	Intro to Contemporary Dance	3.00	A	12.00	
MATH	453	Elementary Theory of Numbers	3.00	A	12.00	
MATH	595	Exponential Theory of Numbers	4.00	A	16.00	
PHYS	211	University Physics: Mechanics	4.00	PS	0.00	

********** CONTINUED ON NEXT COLUMN **********

UCLA CS132

UCLA CS 180

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:						
Dept proficiency test						
	Ehrs: 17.00	GPA-Hrs: 13.00	QPts: 52.00	GPA: 4.00		
Deans List						
Spring 2020 - Urbana-Champaign						
Grainger Engineering						
Computer Science						
CS	242	Programming Studio	3.00	CR	0.00	
CS	421	Progrmg Languages & Compilers	3.00	A	12.00	
	Ehrs: 6.00	GPA-Hrs: 3.00	QPts: 12.00	GPA: 4.00		

********** TRANSCRIPT TOTALS **********

	Earned Hrs	GPA Hrs	Points	GPA
TOTAL INSTITUTION	141.00	133.00	522.03	3.92
TOTAL TRANSFER	0.00	0.00	0.00	0.00
OVERALL	141.00	133.00	522.03	3.92

********** END OF TRANSCRIPT **********