CS - 3240 - Introduction to the Theory of Computation

z08. UG Course Modification #2 (Substantive College/School)

Due Dates

Deadlines for curriculum can be found:

Curriculum SharePoint
Curriculum Website
Procedural Calendar
On your Curriculog dashboard under 'My Upcoming Events'

In order to meet the deadline, this proposal must be on the *Substantive College/School Level Review* step on or before the listed due date.

Directions for Form

Please read instructions and information below before you begin your curriculum proposal. You may also consult the following resources which can provide additional assistance in understanding this form and the curriculum process.

Originator How-To Guide
Curriculum SharePoint
Example Proposal

This form SHOULD be used for the following:

Modifying a course without a special designation (General Studies, Service Learning, Multicultural, or Senior Experience).
Converting an omnibus or individual variable topic course to a regular course.

This form SHOULD NOT be used for the following:

Modifying a course with a special designation (General Studies, Service Learning, Multicultural, or Senior Experience).
Creating a new course with or without a special designation (General Studies, Service Learning, Multicultural, or Senior Experience).
Changing a course prefix or changing the ownership of a course.
Modifying or creating graduate courses.

Instructions:

Import the course you wish to modify with the import button at the top of this page.

Fill in the required information in the following fields:

Originator Name and Email
Curriculum Proposal Justification and Resource Implication Subsection
Review for Conflict and Overlap (end of form)

Launch the Proposal

DO NOT MAKE CHANGES TO YOUR PROPOSAL UNTIL AFTER YOU LAUNCH THE PROPOSAL in order to track changes. Failure to use the track changes feature may cause a delay or denial of your proposal.

Modify course components as needed

Approve the proposal

Use the checkmark icon on the right of the screen to approve the proposal.

Additional Information

You may collapse individual sections of this form by clicking the arrow or "V" icon to the right of the section title.
All fields that are marked with an asterisk (*) are required.
Each section may have additional directions attached. Please follow instructions. Proposals that are incomplete or filled out incorrectly will be returned to the originator.

If you have questions or need assistance in filling out this proposal form, you may contact the Director of Curriculum and Catalog.

Department and Originator Information

College/School:* College of Health and Applied Sciences

Department:* Department of Computer Sciences

Name of Proposal Originator* Jody Paul

Email of Proposal Originator* pauljod@msudenver.edu

Curriculum Proposal Justification and Resource Implication

Justification and Rationale for Curriculum Proposal:* Updating the course prerequisites, description, reading, student behavioral learning objectives, and course content outline.

Resource Implication Narrative:* No new resource implications.

Related Curriculum Proposals:* N/A

Please select from the below list all of the course modifications you are making. If you wish to change something that is not listed, please refer to the other course modification forms in Curriculog. Please do not make changes to fields you do not specify are being modified and make sure to list ALL fields that you modify.

Note: This form CANNOT be used to change courses with special designations (Multicultural, Service Learning, Senior Experience, General Studies)

Course Modifications:* Banner Enforced prerequisites, corequisites, pre/corequisites | Outline | Required reading | Student Behavioral Learning Objectives (SBLOs) | Course description | Prerequisites, corequisites, pre/corequisites | Evaluation of student | Grade mode

Impact Report Results:*

Impact Report for CS 3240

Source: 2021-2022 Undergraduate Catalog	
Prerequisite(s) or Corequisite(s):	CS 4250 - Software Engineering Principles
Cross Listed Course(s):	HON 3240 - Introduction to the Theory of Computation
Prerequisite(s):	CS 3210 - Principles of Programming Languages
	CS 4050 - Algorithms and Algorithm Analysis
	CS 4260 - Software Engineering Practices
	HON 4050 - Algorithms and Algorithm Analysis
Programs	Computer Science Major, B.S.
	Senior Experience Graduation Requirements

Reminder: DO NOT MAKE CHANGES TO YOUR PROPOSAL UNTIL AFTER YOU LAUNCH THE PROPOSAL in order to track changes. Failure to use the track changes feature may cause a delay or denial of your proposal.

Course Title Information

Prefix:* CS

Course Number:* 3240

Course Title:* Introduction to the Theory of Computation

Transcript/Banner Course Title:* Intro to Theory of Computation

Course Type:* Computer Science

CIP Code: 11.0701

Course Hours, Restrictions, and Repeat Information

Please check all that apply from the selections below. You may select more than one option if applicable.

Check All that Apply:*

- [x] Required for Major
- [ ] Required for Minor
- [ ] Required for Concentration
- [ ] Required for Certificate
- [ ] Elective
- [ ] Specified Elective

To receive Title IV financial aid funds, all institutions of higher education must comply with the federal definition of a credit hour. The Higher Learning Commission requires institutions to maintain policies and procedures for verifying compliance with this definition.

Federal Credit Hour Definition: *A credit hour is an amount of work represented in intended learning outcomes and verified by evidence of student achievement that is an institutionally-established equivalency that reasonably approximates not less than:*
(1) one hour of classroom or direct faculty instruction and a minimum of two hours of out-of- class student work each week for approximately fifteen weeks for one semester or trimester hour of credit, or ten to twelve weeks for one quarter hour of credit, or the equivalent amount of work over a different amount of time; or (2) at least an equivalent amount of work as required in paragraph (1) of this definition for other activities as established by an institution, including laboratory work, internships, practica, studio work, and other academic work leading toward to the award of credit hours. 34CFR 600.2 (11/1/2010)

Credits:* 2

Distribution of Credits:* (2 + 0)

Schedule Type(s):* Lecture

Grade Mode(s):* Letter

Face-to-Face or Equivalent Hours per course

Consult Appendix B and C of the Curriculum Manual to determine the hours for the course

Lecture: 30

Lab: 0

Internship: 0

Practicum: 0

Other Hours: 0

Additional Student Work Hours: 60

Please answer yes or no to the below questions. If you answer yes to any of the questions, please fill out the related field on the right.

Is this course a specified repeatable course?* ○ No ○ Yes

If yes, indicate specified repeatable number of credits and/or repeats allowed:

A specified repeatable course is a course that allows a student to repeat the course either in its entirety or for a certain identified total number of credit hours. If you decide to make your course repeatable, please specify either how many times a student can repeat the course for credit, or for the total number of credits they can receive.

Is this course a variable topics umbrella course?* ○ No ○ Yes

If yes, indicate variable topic number of credits and/or repeats available:

Are there course ○ No

If yes, list all

Are there course equivalencies?* ○ No ○ Yes

If yes, list all equivalent courses:

A crosslisting is when a course is made available under additional prefixes for students in other programs.

An equivalency is when two courses are coded in Banner to be equal to each other.

Generally equivalencies are used when an old, archived course is needed to be equal to a new course. Crosslistings are used for all active courses. Supporting documentation should be included to demonstrate approval for crosslistings.

Are there course crosslistings?* ○ No ◉ Yes

If yes, list all crosslistings: HON 3240

Registration Restrictions

Program:

Major:

Level:

Class:

Student Attribute:

Catalog Course Information

The following fields will allow you to attach prerequisites, corequisites, or prerequisites or corequisites to your course. Please specify if you want and of these prerequisites, corequisites, or prerequisites or corequisites Banner enforced.

Banner enforcement means that the requirement will be enforced when the student attempts to register for a course. If you do not Banner enforce the requirement, the system will not check the student's record for the requirement to be met.

Please also indicate the minimum passing grade. If you do not indicate a minimum passing grade, it will default to a "D-" and you will be required to complete another curriculum proposal to modify this minimum passing grade, even if your program has a different minimum passing grade.

Prerequisite(s): CS 2050 and CS 2240 each with a grade of "C-" or better, or permission of instructor

Banner Enforced Prerequisite(s): (CS 2050 or CSI 2050 or CSI 2300) and (CS 2240 or MTH 3170 or MTH 3100)

Minimum Passing Grade for Banner Enforced Prerequisite(s): C-

Corequisite(s):

Banner Enforced

Corequisite(s):

Prerequisite(s) or Corequisite(s):

Banner Enforced Prerequisite(s) or Corequisite(s):

Minimum Passing Grade for Banner Enforced Prerequisite(s) or Corequisite(s):

Catalog Course Description:* This course explores these foundations of computing and computer science: languages, computability, and complexity. Types of languages are identified by their expressive power and how they are represented and implemented. Explorations of computability and complexity provide definitive means for determining whether or not a given problem can be solved at all, and if so, how easy or difficult it is to solve.

The note field DOES show up in the course listing in the university catalog. A note should be made in specific instances where additional information about a course needs to be conveyed to students. The most common reasons for adding a note are:

The course is crosslisted Example: ***(Note: Credit will be granted for only one prefix.)***

Variable credit courses Example: ***(Note: Variable Credit)***

A course is repeatable Example: ***(Note: This course may be repeated up to 3 times under different topics)*** **OR** ***(Note: This course is repeatable for a maximum of six semester hours)***

If a student cannot take two courses and earn credit for both Example: ***(Note: Students cannot earn credit for XXX1234 and XXX2345)***

Note: Credit will be granted for only one prefix: CS or HON.

Lab Fees:

Field Trips:

Course Content

The following section is the course content. You must adhere to the following format for each section:

Required reading: **Smith, J.R. (2014). *Book of Examples*. New York, NY: McGraw-Hill**

List each material in this format. If there are multiple materials please format them in a bullet or list style

Specific Measurable Student Behavioral Learning Objectives: **1, a, i, ii, etc.**
Detailed Outline of Course Content or Outline of Field Experience/Internship: **I, A, 1, a, etc.**
Evaluation of Student Performance: **1, a, i, ii, etc.**

You must use the numbering list feature within the toolbar above each field. Right click on a number in the list and select "Numbered List Properties" to change the numbering style to adhere to the above formatting requirements.

Reminder: DO NOT MAKE CHANGES TO YOUR PROPOSAL UNTIL AFTER YOU LAUNCH THE PROPOSAL in order to track changes. Failure to use the track changes feature may cause a delay or denial of your proposal.

Required reading and other materials will be equivalent to:*

Sipser, M. (2012). Introduction to the Theory of Computation (Third Edition), Boston, MA: Cengage Learning

Critchlow, C. and Eck, D. (2011). Foundations of Computation. N/A: Creative Commons, https://open.umn.edu/opentextbooks/textbooks/foundations-of-computation

Rodger, S. H. and Finely, T. W. (2006) JFLAP: An Interactive Formal Languages and Automata Package. Sudbury, MA: Jones & Bartlett
Publishers, http://www.jflap.org/jflapbook/jflapbook2006.pdf

Specific, Measurable Student Behavioral Learning Objectives:*

Upon completion of this course the student should be able to:

1. Identify properties of formal languages.
2. Discuss the Chomsky language hierarchy (including: regular languages, context-free languages, decidable languages, and recognizable languages) and arrange classes of languages in order of increasing generality.
3. Represent languages using set-theoretic notation, mathematical expressions, grammars, automata, and Turing machines.
4. Represent regular languages as DFA, NFA, Regular Expressions, Regular Grammars, and use set-theoretic notation and interpret such representations.
5. Represent context-free languages as PDA and CFG/CNF and interpret such representations.
6. Determine if a given string is in a language specified by any of the representations.
7. For a given grammar and specified string, produce the derivation(s) and corresponding parse trees.
8. Demonstrate that a language is regular, not-regular but context free, not-context-free but decidable, not-decidable but recognizable, or not-recognizable.
9. Trace the computation of a given Turing machine.
10. Create a Turing machine to solve a specified problem.
11. Determine whether a problem is solvable (decidable or undeciable).
12. Discuss problem tractability, including reference to P vs NP.

Detailed Outline of Course Content (Major Topics and Subtopics) or Outline of Field Experience/Internship *

1. Formal Languages and Representations
    1. Alphabets
    2. Strings
    3. Languages
    4. Grammars
        1. Derivations
        2. Parse Trees
2. Regular Languages
    1. Definition
    2. Deterministic Finite Automata
    3. Nondeterministic Finite Automata
    4. Regular Grammars
    5. Regular Expressions
3. Non-Regular Languages
    1. Proof of non-regularity
4. Context-Free Languages
    1. Definition
    2. Context-Free Grammars
    3. Push-Down Automata
    4. Ambiguity (Grammar and Language)
    5. Proof of non-Context-Free
5. Turing Machines
    1. Turing Machine Conceptualization
    2. Computing with Turing Machines
    3. Turing Machine Extensions
        1. Nondeterministic
        2. Multi-tape
        3. Random access
    4. Universal Turing Machines
6. Computability
    1. Undecidabillity
    2. Church-Turing Thesis
    3. Halting Problem
    4. Unsolvable Problems
    5. Recursive Language Properties

7. Complexity

    1. Asymptotic Notation
    2. Tractability
    3. P versus NP

Evaluation of Student Performance:*

1. One or more of the following

    1. In-class activities
    2. Homework assignments
    3. Quizzes
    4. Examinations

Review for Conflict and Overlap

According to the Undergraduate Curriculum Manual, it is the responsibility of both the originator as well as each level of review to consider potential overlap and curriculum conflict. Any potential overlap or conflict with existing curriculum should be reviewed, and the impacted department(s) should be requested to provide a letter of notification or support, depending on the circumstances. Full information on overlap/conflict can be found here.

Attach documentation that supports affected Departments were notified and/or provided support of the proposed changes in the Proposal Toolbox by clicking on the paperclip icon on the right side of the form.

Please Confirm That:* ☑ I, the originator of this proposal, have completed the necessary due diligence to review this proposal for any potential overlap and/or conflict with existing curriculum. Any departments identified as having potential overlap and/or conflicts have been contacted and a letter of notification and/or a letter of support has been obtained.

Academic Affairs and Registrar's Office Use Only

Notes

Director added grade mode and evaluation to specify course modification field.

This course modification will be effective for the University 2022-2023 Undergraduate Catalog and will be reflected in Banner beginning in Fall 2022.