

CS - 1050 - Computer Science 1

04. UG Course Modification (No Special Designation)

Due Dates and Resources

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

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 *Nonsubstantive, Substantive Colleg/School, OR Substantive University Level Review* step on or before the listed due date. Which step will depend on the changes being made.

Resources for curriculum can be found:

Originator How-To
Guide
Curriculum
SharePoint

This form **SHOULD** be used for the following:

Modification of a course **without** a special designation (General Studies, Service Learning, Ethnic Studies & Social Justice, or Senior Experience).

This form **SHOULD NOT** be used for the following:

Modifying a course with a special designation (General Studies, Service Learning, Ethnic Studies & Social Justice, or Senior Experience).
Creating a new course with or without a special designation (General Studies, Service Learning, Ethnic Studies & Social Justice, or Senior Experience).
Creating or

modifying a
graduate course.

Directions for Form

General Instructions and 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.

Note: Proposals that are incomplete or filled out incorrectly will be returned to the originator.

INSTRUCTIONS FOR MODIFYING AN EXISTING COURSE

Import the course you wish to modify.

Fill out Part I of the form.

Carefully follow the instructions on selecting the level of review your proposal requires.
Incorrect review selection will require a resubmission of the proposal.

LAUNCH the proposal.

Fill out Part II to indicate all the modifications you make to the course.

Modify any course fields in Part III as needed.

Modification Note: DO NOT MAKE CHANGES TO YOUR COURSE

INFORMATION (PART III) 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.

If you modify additional fields not already indicated in Part II, please make sure you add them to the modification list.

Approve the proposal.

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

Part I: Department and Originator Information (Fill out BEFORE launching the proposal)

If you are changing course ownership, please list both departments and if applicable, both college/schools.

College/School:*

College of Aerospace, Computing, Engineering, and Design

Department:*

Department of Computer Sciences

**Name of Proposal
Originator*** Daniel Pittman

**Email of Proposal
Originator*** dpittma8@msudenver.edu

**Justification and
Resource Implication
for Curriculum
Proposal:***

Proposed Change:

Removal of the "Readiness for MTH 1110 - College Algebra for Calculus" as a prerequisite for CS 1050 - Computer Science 1. Likewise, the use of CS1030 to satisfy the mathematics prerequisite is no longer appropriate.

Rationale:

The curriculum committee, in collaboration with the Computer Sciences Department, has determined that the requirement for "Readiness for MTH 1110" is not essential for success in CS 1050. This prerequisite removal is proposed based on the understanding that all necessary mathematical concepts are introduced and taught within the course itself.

Detailed Considerations Supporting This Change:

Integrated Instruction:

- **Inclusive Learning Environment:** CS 1050 is structured to introduce mathematical concepts through in-class examples and discussions that do not assume prior mathematical background. This approach ensures that all students, regardless of their previous mathematical preparation, can access and engage with the course content effectively.
- **Accessible Content Delivery:** By designing course material to be universally accessible, CS 1050 fosters an inclusive learning atmosphere where mathematical concepts are presented in a context that is understandable and relevant to all students. This method supports the department's commitment to diversity and inclusivity by making foundational computer science education approachable for a broader audience.

Learning Outcomes Alignment:

- **Comprehensive Curriculum Design:** The curriculum of CS 1050 has been created to ensure that all necessary mathematical concepts are integrated within the course itself. This integrated approach means that students do not need prior readiness for MTH 1110 to succeed, as the course is structured to develop these skills alongside core programming competencies.
- **Emphasis on Core Computing Skills:** CS 1050 focuses on developing students' abilities in logical reasoning, problem-solving, and understanding the fundamentals of software engineering. These critical skills are taught in a way that does not require prior knowledge of College Algebra, ensuring that all students can fully engage with the course material and achieve the intended learning outcomes.

Accessibility and Inclusivity:

- **Enhancing Access:** Removing the "Readiness for MTH 1110" prerequisite reduces barriers to entry, making CS 1050 more accessible to a wider array of students, including those without a strong background in mathematics.
- **Broadening Participation:** This change is aimed at increasing the diversity of students enrolled in the course by welcoming individuals who might otherwise be deterred by the current prerequisite.

Resource Implications:

- **Advising and Guidance:** Academic advisors will update advising materials to reflect this change, ensuring students are well-informed about their eligibility to enroll in CS 1050 without the need for readiness in College Algebra.
- **Promotional Adjustments:** Minimal updates will be required for promotional materials to accurately reflect the streamlined entry path into the course.

**Related Curriculum
Proposals:*** N/A

Impact Report for CS 1050

Source: 2024-2025 Undergraduate Catalog	
Description:	CS 2050 - Computer Science 2
Prerequisite(s):	CS 2050 - Computer Science 2
	CS 2400 - Computer Organization 2
	CSS 2227 - The C Programming Language
	CSS 2257 - The C++ Programming Language
	CSS 2425 - Introduction to UNIX
	CSS 2751 - Principles of Cybersecurity
	CSS 2752 - Information Assurance
	CSS 3607 - Advanced Web Programming: Java and Perl
	CSS 3707 - Advanced Multimedia Programming
	MTH 4480 - Numerical Analysis I
	MTR 3040 - Computer Programming for Meteorologists
Programs	Advanced Manufacturing Sciences Major, B.S.
	American Chemical Society Post-Baccalaureate Bridge Program (ACS-BP)
	Biochemistry Major, B.S.
	Biology Major, B.S.
	Civil Engineering Technology Major, B.S.
	Computer Engineering Major, B.S.
	Computer Science Major, B.S.
	Computer Science Minor
	Computer Security Minor
	Cybersecurity Major, B.S.
	Data Science and Machine Learning Major, B.S.
	Elementary Education Licensure [INACTIVE]
	Geospatial Sciences Major, B.S. [DISCONTINUED FALL 2021]
	Mathematics Major, B.S.
	Mathematics Minor
	Mechanical Engineering Technology Major, B.S.
	Meteorology Major, B.S.
	Secondary Education Licensure [inactive old]
	Senior Experience Graduation Requirements
	Statistics Minor

Course Modification Level Review

The modification level review question will determine the workflow of this proposal, so it is essential you select the correct one. Please consult this tool which will help you determine the review level you need. If you select a level of review that does not include the changes you make to the course, this proposal will be denied and you will have to resubmit a new proposal. If you are unsure which level of review you need to select, please contact the Curriculum Staff [here](#).

Note: Changing the level of review after launching the proposal will not change the workflow. If you discover that you have selected an incorrect review level post-launch, you must submit a new proposal. You can contact the Curriculum Office to delete the incorrect proposal.

Course Modification Level Review Selection:* Nonsubstantive Substantive College/School Level Substantive University Level

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. 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.

Part II: Course Modification Information

Reminder: This form CANNOT be used to change courses with special designations (Ethnic Studies & Social Justice, Service Learning, Senior Experience, General Studies). If you submit a course with special designation(s) with this form, it will be denied and you will need to resubmit.

Please indicate on the below list all of the course modifications you are making. **Please do not make changes to sections you do not specify are being modified and make sure to select ALL sections that you modify.**

Specify which sections have been modified (check all that apply):

- Part IIIa (prefix, course number, course title, transcript/banner course title, course type, CIP code)
- Part IIIb (credits, distribution of credits, schedule type, grade mode, contact hours, repeats, equivalencies, crosslistings, registration restrictions)
- Part IIIc (prerequisites, corequisites, or prerequisite(s)/corequisite(s), banner enforced prerequisites, corequisites, or prerequisite(s)/corequisite(s), catalog course description, catalog note, lab fees, field trips)
- Part IIIId (required reading, SBLOs, outline, evaluation of student performance)

If the course modification includes changing prefixes, is it a new prefix? Yes No N/A

Reminder: DO NOT MAKE CHANGES TO YOUR COURSE INFORMATION (PART IIIa-III d) 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.

If you modify additional sections not already indicated, please make sure you add them to the modification checklist.

Part IIIa: Course Information

Prefix:*

CS

Course Number:* 1050

Course Title:* Computer Science 1

Transcript/Banner Course Title:* Computer Science 1

Course Type:*

Computer Science

CIP Code: 11.0701

Part IIIb: Course Information, continued

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

- Check All that Apply:*
- 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:* 4

Distribution of Credits:* 4+0

Schedule Type(s):*

Lecture

Grade Mode(s):*

Letter

Pass/Fail

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: 50

Lab:

Internship:

Practicum:

Other Hours:

Additional Student Work Hours: 120

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.

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 specified repeatable course? * No Yes

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

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

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

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
equivalencies? * No
 Yes

If yes, list all
equivalent courses in
alphabetical order:

Are there course
crosslistings? * No
 Yes

If yes, list all
crosslistings in
alphabetical order:

Registration Restrictions

Program:

Major:

Level:

Class:

Student Attribute:

Part IIIc: Course Information, continued

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.

Prerequisite(s):

Banner Enforced
Prerequisite(s):

Minimum Passing
Grade for Banner
Enforced
Prerequisite(s):

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:*** Students learn a modern programming language and develop the foundational skills needed to analyze problems and construct programs that solve them. Emphasis is placed on algorithm development, ensuring correctness, and adopting good programming practices. Additionally, students are introduced to the fundamentals of software engineering and the software development life cycle.

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:

Lab Fees:

Field Trips:

Part IIIId: Course Information, continued

The following section is the course content.

Required reading: Please list materials in preferred citation style (eg. MLA, APA, etc.).

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

Specific Measurable Student Behavioral Learning Objectives: Please list the SBLOs in your preferred numbering or bulleting style. Start section with: Upon completion of this course, the student should be able to:.

Detailed Outline of Course Content or Outline of Field Experience/Internship: Please list the course outline in your preferred numbering or bulleting style. It is recommended that you use a numbering format for this field.

Evaluation of Student Performance: Please list the evaluation of student performance in your preferred numbering or bulleting style.

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. Please maintain consistency in the selected numbering or bulleting styles.

Reminder: DO NOT MAKE CHANGES TO YOUR COURSE INFORMATION (PART IIIa-III d) 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.

If you modify additional sections not already indicated, please make sure you add them to the modification checklist.

Required reading and other materials will be equivalent to:*

- Horstmann, Cay. Java Concepts: Early Objects. 8th Edition. ISBN: 978-1-119-62623-7. 2019.
- Ericson, Barbara. AP CS A Java Course. CS Awesome. 2021.
- Eck, David. Introduction to Programming Using Java. 8th Edition. Open Textbook Library. 2015.

**Specific, Measurable
Student Behavioral
Learning Objectives:***

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

1. Write and run a computer program that correctly solves a problem in the range from simple to medium difficulty.
2. Appropriately document a computer program.
3. Use modularity when writing programs.
4. Declare and define classes, methods, and variables.
5. Declare and utilize parameters and return values.
6. Utilize expressions, assignment, decision structures, and looping.
7. Use appropriate data types including integers, real numbers, characters, Booleans, arrays, and strings.
8. Write interactive programs and programs that use text files for input and output.
9. Utilize the top-down problem solving technique and stepwise refinement.
10. Determine the scope and visibility of an identifier.
11. Utilize testing and debugging techniques.
12. State the basic steps of the software life cycle.

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

1. Computers and Programs

1. overview of computer systems, language translating, and development environments
2. algorithms, syntax, semantics, programs and subprograms

2. Software Engineering

1. problem definition
2. modularity
3. top-down design, step-wise refinement
4. object-oriented design, class design
5. software documentation
6. software engineering life cycle

3. Testing

1. error types and detection
2. debugging
3. exceptional conditions

4. Data Types, Variables, and Identifiers

1. integer, real, character, Boolean, string
2. finite precision errors
3. representation
4. scope and visibility
5. constants
6. operators, expressions, and operator precedence

5. Input/Output

1. interactive
2. reading and writing text files
3. recognizing end of file

6. Classes

1. definitions of classes, methods, and objects
2. standard libraries
3. method arguments and return values

7. Decision Structures

1. conditional operators and logical expressions
2. if-then else, nested if-then else
3. case structures

8. Looping

1. while, do while, for loops
 2. infinite loops
9. Arrays
1. one and multi-dimensional arrays
 2. processing using arrays including partially filled arrays
 3. searching - linear
 4. sorting -- selection

Evaluation of Student Performance:*

1. Homework and programming assignments
2. Quizzes and examinations
3. Final examination
4. Research papers and/or Book reports
5. Oral presentations

As determined by the instructor. Written communication skills will be applied in this course.

CAEPD and Registrar's Office Use Only
--

Notes Director made two slight changes to the description to keep it present tense.

This course modification will be effective for the University 2025-2026 Undergraduate Catalog and will be reflected in Banner beginning in Fall 2025.

Form updated May 2024