Metropolitan State University of Denver
Regular Course Syllabus

CS - 1030 - Computer Science Principles Fall 2016

Status	completed
Approval Process Name	08. UG Course Modification #2 (Substantive College) (17-18)
Department	Mathematical and Computer Sciences, Department of
Prefix:	CS
Course Number:	1030
Course Type:	Computer Science
Course Title:	Computer Science Principles
Transcript Course Title:	Computer Science Principles
Check All That Apply:	Required for Major, Elective
Credit Hours:	4
Schedule Type:	Lecture
Grade Mode:	Letter
Lecture:	60
Lab:	
Internship:	
Practicum:	
Other:	
Additional Student Work Hours per course:	120
Variable topics umbrella course:	No
If yes, number of credits/ repeats allowed	
Specified repeatable course:	No
If yes, number of credits/ repeats allowed	
Prerequisite(s):	
Corequisite(s):	
Prerequisite(s) and/or Corequisite(s):	
Banner Prerequisite(s):	
Banner Corequisite(s):	
Banner Prerequisite(s) and/or Corequisite(s):	
Level	
Class	
Program/Major	
Student attribute	
Catalog Course Description:	Computer Science Principles introduces students to the central ideas of computer science vital for success in today's world. Students are invited to develop the computational thinking skills that apply across disciplines, as we explore computing from multiple perspectives, including: *cognitive, economic, ethical, legal, mathematical, philosophical, social,* and *technical*. The course integrates computational thinking practices with big ideas in computing to address: *collaborative teamwork, communication, creativity, critical thinking, innovation, problem solving,* and *programming*. Students are afforded the

	opportunity to participate in active-learning experiences and to create computational artifacts that bring ideas to life.
Required Reading and Other Materials will be equivalent to:	Schneider, G. M. & Gersting, J. (2015) Invitation to Computer Science, 7th Edition. Course Technology. and/or Walker, H. M. (2012) The Tao of Computing, Second Edition, Chapman and Hall/CRC. and/or Parsons, J. J. (2016) New Perspectives on Computer Science, 19th ed. Course Technology. Additional topic-specific readings, such as: • Leiber, J. (1985) Can Animals and Machines Be Persons?: A Dialog. Hackett. • MacCormick, J. & Bishop, C. (2013) Nine Algorithms that Changed the Future. Princeton University Press. Programming Support (language decided by instructor), such as: • MIT Media Lab. (2016) Scratch, https://scratch.mit.edu (accessed: 2 May 2016) • Code.org. (2016) Learn. http://code.org/learn/beyond (accessed: 2 May 2016) • Processing Foundation. (2016) Processing – Tutorials. http://processing.org/tutorials/ (accessed: 2 May 2016)
Specific, Measurable Student Behavioral Learning Objectives:	Upon completion of this course the student should be able to: 1. Analyze effects of computation from multiple perspectives (including, but not limited to, scientific, sociological, ethical, economic, cultural, community, and global) 2. Create computational artifacts 3. Use abstractions and models 4. Evaluate problems and artifacts from computational-inclusive viewpoints 5. Communicate processes and results 6. Work effectively in collaborative teams
Detailed Outline of Course Content (Major Topics and Subtopics) or Outline of Field Experience/ Internship	Course content integrates big ideas in computing with computational thinking practices, both outlined below, such that all practices and big ideas are represented. [Reference: The College Board. (2016) *AP Computer Science Principles: Course and Exam Description.* New York, College Board.] I. Big Ideas in Computing

A. Creativity: Computing is a creative human activity

- Creative development can be an essential process for creating computational artifacts
- Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem
- Computing can extend traditional forms of human expression and experience

B. Abstraction: Abstraction reduces information and detail to facilitate focus on relevant concepts

- A variety of abstractions built on binary sequences can be used to represent all digital data
- Multiple levels of abstraction are used to write programs or create other computational artifacts
- Models and simulations use abstraction to generate new understanding and knowledge

C. Data and Information: Data and information facilitate the creation of knowledge

- People use computer programs to process information and to gain insight and knowledge
- Computing facilitates exploration and the discovery of connections in information
- There are trade-offs when representing information as digital data

D. Algorithms: Algorithms are used to develop and express solutions to computational problems

- Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages
- Algorithms can solve many, but not all, computational problems

E. Programming: Programming enables problem-solving, human expression, and creation of knowledge

- Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)
- People write programs to execute algorithms
- Programming is facilitated by appropriate abstractions

Programs are developed, maintained, and used by people for different purposes

Programming uses mathematical and logical concepts

F. The Internet: The Internet pervades modern computing

The Internet is a network of autonomous systems

Characteristics of the Internet influence the systems built on it

Cybersecurity is an important concern for the Internet and systems built on it

G. Global Impact: Computing has global impact

Computing enhances communication, interaction, and cognition

Computing enables innovation in nearly every field

Computing has global effects — both beneficial and harmful — on people and society

Computing innovations influence and are influenced by the economic, social, and cultural contexts in which they are designed and used

Appropriate technologies and tools facilitate the accessing of information and enable the ability to evaluate the credibility of sources

II. Computational Thinking Practices

A. Connecting Computing

Identify impacts of computing

Describe connections between people and computing

Explain connections between computing concepts

B. Creating Computational Artifacts

Create a computational artifact with a practical, personal, or societal intent

Select appropriate techniques to develop a computational artifact

Use appropriate algorithmic and information management principles

C. Abstracting

	Explain how data, information, and knowledge are represented for computational use Explain how abstractions are used in computation or modeling Identify abstractions Describe modeling in a computational context D. Analyzing Problems and Artifacts Evaluate a proposed solution to a problem Locate and correct errors Explain how and artifact functions Justify appropriateness and correctness of a solution, model, or artifact E. Communicating Explain the meaning of a result in context Describe computation with accurate and precise language, notations, or visualizations Summarize the purpose of a computational artifact F. Collaborating Collaborate with others in solving a computational problem Collaborate with others in producing an artifact Foster a constructive, collaborative climate by resolving conflicts and facilitating the contributions of team members Exchange knowledge and feedback with team members Review and revise collective work as needed to create a high-quality artifact
Evaluation of Student Performance	Evaluation mechanisms may include, but are not limited to, any or all of the following: • In-class Activities • Homework • Projects • Quizzes / Exams
Learning Objectives	

Distribution of Credit Hours	(4 + 0)		
Steps	**Decision**	**Date**	
Originator			
Jody Paul	approved	11/22/2016 12:50PM	
Department Curriculum Committee Chair			
Clark Dollard	approved	11/27/2016 10:22PM	
Department Chair			
Lindsay Packer	approved	11/28/2016 04:16PM	
Dean's Office Tracking Assignment			
Kelsey Smith	approved	11/29/2016 08:39AM	
Substantive College Level			
Jody Paul	approved	06/04/2017 02:19PM	
Linda Lang-Peralta	approved	12/11/2017 05:41PM	
Mona Mocanasu	approved	12/07/2017 06:10PM	
AVP Academic and Student Affairs			
Chad Harris	approved	01/17/2018 08:11AM	