Cerro Coso College

Course Outline of Record Report

10/13/2021

CSCIC255 : Discrete Structures

General Information

Author:	-
Course Code (CB01) :	CSCIC255
Course Title (CB02) :	Discrete Structures
Department:	Business Information Technolog
Proposal Start:	Fall 2013
TOP Code (CB03) :	(0706.00) Computer Science (transfer)
SAM Code (CB09) :	Non-occupational
Distance Education Approved:	Yes
Course Control Number (CB00) :	CCC000547179
Curriculum Committee Approval Date:	05/03/2013
Board of Trustees Approval Date:	06/13/2013
External Review Approval Date:	07/18/2013
Course Description:	This course covers mathematical topics essential for work in computer science, including number bases, mathematical induction, sets, functions, recursion, probability, graphs, trees, logic, Boolean algebra, and proof techniques.
Submission Type:	New Course
Author:	No value

Faculty Minimum Qualifications

Master Discipline Preferred:	• Computer Science
Alternate Master Discipline Preferred:	No value
Bachelors or Associates Discipline Preferred:	No value
Additional Bachelors or Associates Discipline Preferred:	No value

Course Development Options

Basic Skills Status (CB08)

Course is not a basic skills course.

Course Special Class Status (CB13)

Course is not a special class.

Grade Options

- Letter Grade Methods
- Pass/No Pass

☐ Allow Students to Gain Credit by Exam/Challenge

Allowed Number of Retakes

0

Course Prior To College Level (CB21)

Not applicable.

Rationale For Credit By Exam/Challenge

Retake Policy Description

☑ Allow Students To Audit Course

No value

Type:|Non-Repeatable Credit

Course Support Course Status (CB26)

No value

Associated Programs

☑ Course is part of a program (CB24)

Associated Program	Award Type	Active
CC Liberal Arts: Mathematics & Science	A.A. Degree Major	Summer 2018 to Fall 2020
Liberal Arts: Mathematics & Science Associate in Arts Degree	A.A. Degree Major	Fall 2020

Transferability & Gen. Ed. Options

Course General Education Status (CB25)

No value

Transferability

Transferable to both UC and CSU

Transferability Status

Approved

Units and Hours:

Summary

Minimum Credit Units (CB07)	3
Maximum Credit Units (CB06)	3
Total Course In-Class (Contact) Hours	54
Total Course Out-of-Class Hours	108
Total Student Learning Hours	162
Faculty Load	0

Credit / Non-Credit Options

Course Credit Status (CB04)	Course Non Credit Category (CB22)	Non-Credit Characteristic
Credit - Degree Applicable	Credit Course.	No Value

Course Classification Status (CB11)

Credit Course.

Funding Agency Category (CB23)

Not Applicable.

☐ Cooperative Work Experience Education Status (CB10)

☐ Variable Credit Course

Weekly Student Hours

	In Class	Out of Classs
Lecture Hours	3	6
Laboratory Hours	0	0
Activity Hours	0	0

Course Student Hours

Course Duration (Weeks)	18
Hours per unit divisor	0
Course In-Class (Contact) Hours	
Lecture	0
Laboratory	0
Activity	0
Total	54
Course Out-of-Class Hours	
Lecture	0
Laboratory	0
Activity	0
Total	108

Time Commitment Notes for Students

No value

Faculty Load

Extra Duties: 0

Faculty Load: 0

Units and Hours: - Weekly Specialty Hours

Activity Name	Type	In Class	Out of Class
No Value	No Value	No Value	No Value

Pre-requisites, Co-requisites, Anti-requisites and Advisories

Prerequisite

CSCIC252 - Introduction to Computer Science

Basic understanding of computer architecture, binary numbers, and basic programming skills. All of which are taught in the CSCI 252 course.

Entrance Skills

Entrance Skills	Description
No value	No value

Limitations on Enrollment

Limitations on Enrollment	Description
No value	No value

Specifications

Methods of Instruction

Methods of Instruction	Skills Development and Performance
Rationale	No value
Methods of Instruction	Problem Solving
Rationale	No value
Methods of Instruction	Outside reading
Rationale	No value
Methods of Instruction	Laboratory
Rationale	No value
Methods of Instruction	Lecture
Rationale	No value
Methods of Instruction	Instruction through examination or quizzing
Rationale	No value

Methods of Instruction	Computational Work
Rationale	No value

Methods of Instruction	Demonstration
Rationale	No value

Assignments

Reading Text - Preparing for class by reading the chapters assigned and supplemental information
Homework Assignments- Problem sets as handouts or from the text to practice concepts, preparing for presenting topics on the course
Group work - Group work time for a group project

Methods of Evaluation	Rationale
Project	Projects and Problem sets to demonstrate student's skill in graphs and trees, mathematical inductions, proofs, combinatorics. Example: Using this graph, determine the best path.
Tests	Objective tests/quizzes demonstrating student's knowledge of number conversions, mathematical induction, recursion, sets, relations, and functions. Examples: Convert this function to a recursive function.

Equipment

No Value

Textbooks

Author	Title	Publisher	Date	ISBN
	Stein, C. L., Drysdale, R., Bogard, K.. (2011) Discrete Mathematics for Computer Scientists, , Pearson			

Other Instructional Materials

No Value

Materials Fee

No

Learning Outcomes and Objectives

Course Objectives

No value

CSLOs

Describe how formal tools of symbolic logic are used to model real-life situations, including those arising in computing contexts such as program correctness, database queries, and algorithms. Expected SLO Performance: 70.0

Illustrate the basic terminology, properties, and traversal methods for graphs and trees. Expected SLO Performance: 70.0

Apply the binomial theorem to independent events and Bayes' theorem to dependent events. Expected SLO Performance: 70.0

Relate the ideas of mathematical induction to recursion and recursively defined structures. Expected SLO Performance: 70.0

Solve problems involving sets, relations, functions, congruences, and combinatorics. Expected SLO Performance: 70.0

Science Liberal Arts: Mathematics & Science AA Degree	Apply algebraic, graphical, numerical, and other methods to solve applied problems in the areas of mathematics, natural sciences, computer graphics, and computer animation.

Analyze a problem to create relevant recurrence equations. Expected SLO Performance: 70.0

Outline

Course Outline

A. Number bases
a. Binary
b. Hexadecimal
c. Octal
d. Conversions between
B. Sets; relations; functions; congruences
a. Sets including Venn diagrams
b. Complements
c. Operations
d. DeMorgan&rsquo:s law
e. Relations
f. Equivalence classes
g. Functions
i. Injective
ii. Surjective
iii. Inverse
iv. Domain
v. Range
h. Pidgeonhole principle
i. Cardinality and countability
C. Graphs and Trees
a. Directed graphs
b. Undirected graphs

c. Weighted graphs
d. Eulerian and Hamiltonian circuits
e. Traveling sales person
f. Graph coloring
g. Trees
h. Expression trees
i. Tree traversals
D. Logic and Boolean Algebra
a. Truth tables
b. Logical connectivities
c. Propositional calculus
d. Boolean algebra and Boolean circuits
e. Normal form
f. Predicate logic
g. Limitations of predicate logic
E. Computer architecture
a. Overview and history of computer architecture
b. Fundamental building blocks
i. Logic gates
ii. Flip flops
c. Logic expressions; minimization; sum of product forms.
F. Recursion
a. Recursive definitions of functions
b. Factorials
c. Fibonacci sequences
d. Towers of Hanoi
G. Mathematical induction
a. Mathematical induction
b. Strong induction
H. Proof Techniques
a. Direct proof
b. Proffs by counter example; contrapositive and contradiction
c. Logical equivalence
I. Combanitorics
a. Binomials
b. Counting arguments
c. Combinations and permutations
d. Pidgeon-hole principle
e. Solving recurrence relations
J. Discrete probability
a. Finite probability space; measures; evetns
b. Conditional probability
i. Bayes&rsquo: theorem
c. Integer random variables; expectation
A. Number bases
a. Binary
b. Hexadecimal
c. Octal
d. Conversions between
B. Sets; relations; functions; congruences
a. Sets including Venn diagrams
b. Complements
c. Operations
d. DeMorgan&rsquo:s law
e. Relations
f. Equivalence classes
g. Functions
i. Injective
ii. Surjective
iii. Inverse
iv. Domain
v. Range
h. Pidgeonhole principle
i. Cardinality and countability
C. Graphs and Trees
a. Directed graphs
b. Undirected graphs

c. Weighted graphs
d. Eulerian and Hamiltonian circuits
e. Traveling sales person
f. Graph coloring
g. Trees
h. Expression trees
i. Tree traversals
D. Logic and Boolean Algebra
a. Truth tables
b. Logical connectivities
c. Propositional calculus
d. Boolean algebra and Boolean circuits
e. Normal form
f. Predicate logic
g. Limitations of predicate logic
E. Computer architecture
a. Overview and history of computer architecture
b. Fundamental building blocks
i. Logic gates
ii. Flip flops
c. Logic expressions; minimization; sum of product forms.
F. Recursion
a. Recursive definitions of functions
b. Factorials
c. Fibonacci sequences
d. Towers of Hanoi
G. Mathematical induction
a. Mathematical induction
b. Strong induction
H. Proof Techniques
a. Direct proof
b. Proofs by counter example; contrapositive and contradiction
c. Logical equivalence
I. Combanitorics
a. Binomials
b. Counting arguments
c. Combinations and permutations
d. Pidgeon-hole principle
e. Solving recurrence relations
J. Discrete probability
a. Finite probability space; measures; evetns
b. Conditional probability
i. Bayes&rsquo: theorem
c. Integer random variables; expectation

Delivery Methods and Distance Education

Delivery Method: Please list all that apply -Face to face -Online (purely online no face-to-face contact) -Online with some required face-to-face meetings (“Hybrid”) -Online course with on ground testing -iTV – Interactive video = Face to face course with significant required activities in a distance modality -Other

Face 2 Face
Online
Hybrid
Interactive

Rigor Statement: Assignments and evaluations should be of the same rigor as those used in the on-ground course. If they are not the same as those noted in the COR on the Methods of Evaluation and out-of-class assignments pages, indicate what the differences are and why they are being used. For instance, if labs, field trips, or site visits are required in the face to face section of this course, how will these requirements be met with the same rigor in the Distance Education section?

No Value

Effective Student-Instructor Contact: Good practice requires both asynchronous and synchronous contact for effective contact. List the methods expected of all instructors teaching the course. -Learning Management System -Discussion Forums -Moodle Message -Other Contact -Chat/Instant Messaging -E-mail -Face-to-face meeting(s) -Newsgroup/Discussion Board -Proctored Exam -Telephone -iTV -Interactive Video -Other (specify)

contact_moodle_forums
contact_moodle_message
contact_email
contact_phone
contact_itv
contact_other

Software and Equipment: What additional software or hardware, if any, is required for this course purely because of its delivery mode? How is technical support to be provided?

No Value

Accessibility: Section 508 of the Rehabilitation Act requires access to the Federal government's electronic and information technology. The law covers all types of electronic and information technology in the Federal sector and is not limited to assistive technologies used by people with disabilities. It applies to all Federal agencies when they develop, procure, maintain, or use such technology. Federal agencies must ensure that this technology is accessible to employees and the public to the extent it does not pose an "undue burden". I am using -iTV—Interactive Video only -Learning management system -Publisher course with learning management system interface.

s508_itv
s508_moodle
s508_publisher

Class Size: Good practice is that section size should be no greater in distance ed modes than in regular face-to-face versions of the course. Will the recommended section size be lower than in on-ground sections? If so, explain why.

No Value