CIS 10A Course Outline as of Fall 2001

CATALOG INFORMATION

Dept and Nbr: CIS 10A Title: COMP SCI FUNDAMENTALS I
Full Title: Computer Science Fundamentals I
Last Reviewed: 2/8/2021

Units		Course Hours per Week		Nbr of Weeks	Course Hours Total	
Maximum	3.00	Lecture Scheduled	2.00	17.5	Lecture Scheduled	35.00
Minimum	3.00	Lab Scheduled	2.00	8	Lab Scheduled	35.00
		Contact DHR	1.00		Contact DHR	17.50
		Contact Total	5.00		Contact Total	87.50
		Non-contact DHR	0		Non-contact DHR	0

Total Out of Class Hours: 70.00 Total Student Learning Hours: 157.50

Title 5 Category:	AA Degree Applicable
Grading:	Grade Only
Repeatability:	00 - Two Repeats if Grade was D, F, NC, or NP
Also Listed As:	
Formerly:	CIS 10

Catalog Description:

This course introduces computer science and software engineering. Topics include machine architecture, software theory, integrated development environments, object-oriented design and control-of-flow constructs. Eight to ten programs are written in C++. Appropriate as a first course for those pursuing a four-year degree in computer science. Also serves as an introduction to programming for those interested in languages other than C++.

Prerequisites/Corequisites:

Completion of Math 155 or higher.

Recommended Preparation:

Limits on Enrollment:

Schedule of Classes Information:

Description: Introduces computer science & software engineering. Includes machine architecture, software theory, integrated development environments, object-oriented design and control-of-flow constructs. C++ is used. Serves as a first course for four-year degree computer science majors. Also serves as an intro to programming for languages other than C++. (Grade

Only)
Prerequisites/Corequisites: Completion of Math 155 or higher.
Recommended:
Limits on Enrollment:
Transfer Credit: CSU;UC. (CAN CIS 10A+CIS 10A=CSCI22)(CIS 10A+CIS 10A=CSCI22)
Repeatability: Two Repeats if Grade was D, F, NC, or NP

ARTICULATION, MAJOR, and CERTIFICATION INFORMATION:

AS Degree:	**Area**			Effective:	Inactive:
CSU GE:	**Transfer Area**			Effective:	Inactive:
IGETC:	**Transfer Area**			Effective:	Inactive:
CSU Transfer:	Transferable	Effective:	Spring 1989	Inactive:	
UC Transfer:	Transferable	Effective:	Spring 1989	Inactive:	

CID:

CID Descriptor:COMP 122	Programming Concepts and Methodology I
SRJC Equivalent Course(s):	CS10A OR CS10B
CID Descriptor:COMP 112	Introduction to Programming Concepts and Methodologies
SRJC Equivalent Course(s):	CS10A

Certificate/Major Applicable:
Certificate Applicable Course

COURSE CONTENT

Outcomes and Objectives:
Students will:
1. Translate integers, real numbers, and characters into machine representation.
2. Explain machine architectures and the stored-program concept.
3. Compare various computer operating system designs.
4. Create correct and efficient algorithms.
5. Describe alternative programming language designs and implementations.
6. Implement the software-development life cycle.
7. Produce multi-file programs using an integrated development environment.
8. Employ the basic elements of the C++ language.
9. Use object-oriented language features as a client programmer.
10. Implement algorithms using C++ flow-control constructs.
11. Understand career objectives related to Computer Science.
12. Use information and learning resources as they pertain to Computer Science

Topics and Scope:

A. Computer Science
   1. Machine Architecture
      a. Data Storage

1) Primary and secondary storage
2) Coding
a) The binary system
b) Integers, reals and characters
c) Error detection and correction
b. Data Manipulation
1) The CPU
2) The stored-program concept
3) programs as data and their execution
2. Software
a. Operating Systems and Networks
1) History
2) Interprocess coordination
3) Networks
b. Algorithms
1) Representation and discovery
2) Iterative and recursive structures
3) Efficiency and correctness
c. Languages
1) History
2) Traditional concepts
3) Declarative, procedural and object-oriented approaches
d. Software Engineering
1) The software life cycle
2) Tools and techniques
3) Documentation
4) Legal issues
3. Career objectives related to Computer Science
4. Information and learning resources as they pertain to Computer Science
B. Programming
1. Integrated Development Environments
a. Projects
b. Editors
c. Debuggers
d. Browsers
e. Profilers
2. Language Basics
a. Data types and assignments
b. Basic C++ input/output
c. Expressions
d. Flow of control
e. Style
3. Procedural Abstraction
a. Top Down Design
b. Library functions
c. User-defined functions
d. Overloading
4. C++ I/O
a. Streams
b. Manipulators

5. Client Programming and Classes
    a. Structures
    b. Classes
    c. Abstract Data Types
6. Miscellaneous flow of control Constructs
    a. Nesting
    b. Enumerations
    c. The case construct (switch)

Assignment:

1. Maintain a reading schedule for the text(s).
2. Write programs using the C++ programming language.
3. Test and debug programs.
4. Write program documentation.

Methods of Evaluation/Basis of Grade:

Writing: Assessment tools that demonstrate writing skills and/or require students to select, organize and explain ideas in writing.

Written program documentation	Writing 10 - 20%

Problem Solving: Assessment tools, other than exams, that demonstrate competence in computational or non-computational problem solving skills.

Homework problems, Exams, LABORATORY ASSIGNMENTS	Problem solving 20 - 60%

Skill Demonstrations: All skill-based and physical demonstrations used for assessment purposes including skill performance exams.

None	Skill Demonstrations 0 - 0%

Exams: All forms of formal testing, other than skill performance exams.

Multiple choice, True/false, Matching items, Completion, Programming exercises	Exams 20 - 60%

Other: Includes any assessment tools that do not logically fit into the above categories.

None	Other Category 0 - 0%

Representative Textbooks and Materials:
"Problem Solving with C++: The Object of Programming", by Walter Savitch - Addison-Wesley Longman 1999

"Computer Science: An Overview", by J. Brookshear - Addison-Wesley Longman 1997