

CS 10A Course Outline as of Fall 2018**CATALOG INFORMATION**

Dept and Nbr: CS 10A Title: INTRO TO PROGRAMMING
 Full Title: Introduction to Programming Concepts and Methodologies
 Last Reviewed: 2/8/2021

Units		Course Hours per Week		Nbr of Weeks	Course Hours Total	
Maximum	4.00	Lecture Scheduled	3.00	17.5	Lecture Scheduled	52.50
Minimum	4.00	Lab Scheduled	3.00	6	Lab Scheduled	52.50
		Contact DHR	0		Contact DHR	0
		Contact Total	6.00		Contact Total	105.00
		Non-contact DHR	0		Non-contact DHR	0

Total Out of Class Hours: 105.00

Total Student Learning Hours: 210.00

Title 5 Category: AA Degree Applicable

Grading: Grade or P/NP

Repeatability: 00 - Two Repeats if Grade was D, F, NC, or NP

Also Listed As:

Formerly: CS 10

Catalog Description:

Specification, design, implementation, testing, debugging, maintenance, and documentation of computer programs. Topics include algorithms, languages, software engineering, control structures, functions, and arrays. Numerous programs are written in C++. Intended for both computer science majors and for those seeking a general introduction to computer programming.

Prerequisites/Corequisites:**Recommended Preparation:**

Eligibility for ENGL 1A or equivalent; AND completion of MATH 155 and some computer experience

Limits on Enrollment:**Schedule of Classes Information:**

Description: Specification, design, implementation, testing, debugging, maintenance, and documentation of computer programs. Topics include algorithms, languages, software engineering, control structures, functions, and arrays. Numerous programs are written in C++. Intended for both computer science majors and for those seeking a general introduction to

computer programming. (Grade or P/NP)

Prerequisites/Corequisites:

Recommended: Eligibility for ENGL 1A or equivalent; AND completion of MATH 155 and some computer experience

Limits on Enrollment:

Transfer Credit: CSU;UC.

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:

Both Certificate and Major Applicable

Approval and Dates

Version:	011	Course Created/Approved:	3/9/1989
Version Created:	4/17/2017	Course Last Modified:	8/15/2021
Submitter:	Dave Harden	Course last full review:	2/8/2021
Version Status:	Approved (Changed Course)	Prereq Created/Approved:	2/8/2021
Version Status Date:	5/8/2017	Semester Last Taught:	Summer 2021
Version Term Effective:	Fall 2018	Term Inactive:	Fall 2021

COURSE CONTENT

Student Learning Outcomes:

At the conclusion of this course, the student should be able to:

1. Describe the principles of structured programming and be able to describe, design, implement,
and test structured programs using currently accepted methodology.
2. Explain what an algorithm is and its importance in computer programming.

Objectives:

Upon completion of this course students will be able to:

1. Create correct and efficient algorithms.
2. Describe the software-development life cycle.
3. Employ the basic elements of the C++ language.

4. Implement algorithms using C++ flow-control constructs.
5. Write descriptive and helpful program documentation.
6. Implement algorithms using arrays.

Topics and Scope:

- I. Software life-cycle including design, development, styles, documentation, testing and maintenance
- II. Procedural versus objected oriented programming - Survey of Current Languages
- III. Program Design Tools and Programming Environments
- IV. Documentation
- V. Coding Conventions
- VI. Data Types, Variables, Expressions, Sequential Processing
- VII. Arrays
 - A. Declaring and allocating arrays
 - B. Multiple-subscripted arrays
- VIII. Control Structure
 - A. Selective structures: if and switch
 - B. Repetitive structures: loops
- IX. Algorithms including Simple Sorting and Searching
- X. File I/O
 - A. Files and streams
 - B. Sequential access files
- XI. Error Handling
- XII. Passing Parameters by Value and by Reference
- XIII. Principles of Testing and Designing Test Data

All topics are covered in both the lecture and lab parts of the course.

Assignment:

Lecture Related Assignments:

1. Read approximately 30 pages per week
2. Complete 2-8 examinations including final exam

Lab Related Assignments:

1. Complete 10-15 programming assignments, with documentation, using the C++ programming language

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.

Programming 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.

Exams, Final Exam: (Multiple choice, true/false, matching items, completion, programming problems)

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:

Starting Out with C++ From Control Structures through Objects. 8th ed. Gaddis, Tony. Pearson. 2014

OTHER REQUIRED ELEMENTS

STUDENT PREPARATION

Matric Assessment Required:	B	Requires Both English & Math Assessment
Prerequisites-generate description:	NP	No Prerequisite
Advisories-generate description:	U	User-Generated Text
Prereq-provisional:	N	NO
Prereq/coreq-registration check:	N	No Prerequisite Rules Exist
Requires instructor signature:	N	Instructor's Signature Not Required

BASIC INFORMATION, HOURS/UNITS & REPEATABILITY

Method of instruction:	02	Lecture
	04	Laboratory
	72	Internet-Based, Delayed Interaction
	71	Internet-Based, Simultaneous Interaction
Area department:	CS	Computer Studies
Division:	72	Arts & Humanities
Special topic course:	N	Not a Special Topic Course
Program status:	1	Both Certificate and Major Applicable
Repeatability:	00	Two Repeats if Grade was D, F, NC, or NP
Repeat group id:		

SCHEDULING

Audit allowed:	Y	Auditable
Open entry/exit:	N	Not Open Entry/Open Exit
Credit by exam:	N	Credit by examination not allowed
Budget code: Program:	0000	Unrestricted
Budget code: Activity:	0701	Computer & Information Science

OTHER CODES

Discipline:	Computer Science	
Basic skills:	N	Not a Basic Skills Course
Level below transfer:	Y	Not Applicable
CVU/CVC status:	Y	Distance Ed, Not CVU/CVC Developed
Distance Ed Approved:	Y	Exclusively online or other technology based instruction
Emergency Distance Ed Approved:	N	None
Credit for Prior Learning:	N	Agency Exam
	N	CBE
	N	Industry Credentials
	N	Portfolio
Non-credit category:	Y	Not Applicable, Credit Course
Classification:	Y	Liberal Arts and Sciences Courses
SAM classification:	E	Non-Occupational
TOP code:	0706.00	Computer Science (Transfer)
Work-based learning:	N	Does Not Include Work-Based Learning
DSPS course:	N	Not a DSPS Course
In-service:	N	Not an in-Service Course

