

**CS 10 Course Outline as of Fall 2010****CATALOG INFORMATION**

Dept and Nbr: CS 10

Title: INTRO TO PROGRAMMING

Full Title: Introduction to Computer Programming

Last Reviewed: 2/8/2021

| Units   |      | Course Hours per Week |      | Nbr of Weeks | Course Hours Total |       |
|---------|------|-----------------------|------|--------------|--------------------|-------|
| Maximum | 4.00 | Lecture Scheduled     | 4.00 | 17.5         | Lecture Scheduled  | 70.00 |
| Minimum | 4.00 | Lab Scheduled         | 1.00 | 6            | Lab Scheduled      | 17.50 |
|         |      | Contact DHR           | 0    |              | Contact DHR        | 0     |
|         |      | Contact Total         | 5.00 |              | Contact Total      | 87.50 |
|         |      | Non-contact DHR       | 0    |              | Non-contact DHR    | 0     |

Total Out of Class Hours: 140.00

Total Student Learning Hours: 227.50

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: CIS 10

**Catalog Description:**

Specification, design, implementation, testing, debugging, maintenance, and documentation of computer programs. Topics include algorithms, languages, software engineering, control structures, functions, data abstraction using classes, 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:**

Completion of MATH 154 or higher (V2)

**Recommended Preparation:**

Eligibility for ENGL 1A or equivalent

**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, data abstraction using classes, 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: Completion of MATH 154 or higher (V2)  
 Recommended: Eligibility for ENGL 1A or equivalent  
 Limits on Enrollment:  
 Transfer Credit: CSU;UC.  
 Repeatability: Two Repeats if Grade was D, F, NC, or NP

## **ARTICULATION, MAJOR, and CERTIFICATION INFORMATION:**

|                   |                      |            |           |
|-------------------|----------------------|------------|-----------|
| <b>AS Degree:</b> | <b>Area</b>          | Effective: | Inactive: |
| <b>CSU GE:</b>    | <b>Transfer Area</b> | Effective: | Inactive: |

|               |                      |            |           |
|---------------|----------------------|------------|-----------|
| <b>IGETC:</b> | <b>Transfer Area</b> | Effective: | Inactive: |
|---------------|----------------------|------------|-----------|

|                      |              |            |             |           |
|----------------------|--------------|------------|-------------|-----------|
| <b>CSU Transfer:</b> | Transferable | Effective: | Spring 1989 | Inactive: |
|----------------------|--------------|------------|-------------|-----------|

|                     |              |            |             |           |
|---------------------|--------------|------------|-------------|-----------|
| <b>UC Transfer:</b> | 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

## **COURSE CONTENT**

### **Outcomes and Objectives:**

Upon completion of this course students will be able to:

1. Translate integers, real numbers, and characters into machine representation.
2. Create correct and efficient algorithms.
3. Implement the software-development life cycle.
4. Produce multi-file programs using an integrated development environment.
5. Employ the basic elements of the C++ language.
6. Use object-oriented language features as a client programmer.
7. Implement algorithms using C++ flow-control constructs.
8. Identify career objectives related to Computer Science.
9. Use information and learning resources as they pertain to Computer Science.

### **Topics and Scope:**

1. Basic Computer Organization
  - a. Basic hardware components of a computer
  - b. Binary representation of data
2. Algorithms
  - a. The concept of an algorithm
  - b. Problem solving techniques
  - c. Efficiency and correctness
3. Languages
  - a. History

- b. Procedural vs. object-oriented
- c. Compiled vs. interpreted
- 4. Software Engineering
  - a. The software life cycle
  - b. Object-oriented design
  - c. Functional decomposition
  - d. Documentation
  - e. Ethical Issues
- 5. Integrated Development Environments
  - a. Projects
  - b. Editors
  - c. Debuggers
- 6. C++ Language Basics
  - a. Interactive input/output
  - b. File input/output
  - c. Variables and constants
  - d. Arithmetic expressions and operators
  - e. Data types int, double, char, bool, and string
- 7. Decision Structures
  - a. If and if/else statements
  - b. Switch statement
  - c. Logical expressions and operators
- 8. Iteration Structures
  - a. While statement
  - b. Do-while statement
  - c. For statement
  - d. Nested logic
- 9. Functions
  - a. User-defined functions: void and value-returning
  - b. Top-down design/stepwise refinement
  - c. Procedural abstraction
  - d. Scope and lifetime of identifiers
  - e. Reference and value parameters
  - f. Library functions
- 10. Classes
  - a. Structs
  - b. Data abstraction and abstract data types
  - c. Encapsulation/information hiding
  - d. Private/public members
  - e. Class scope
  - f. Using multiple files
  - g. Constructors, including multiple constructors
  - h. Passing objects using const &
  - i. Declaring const member functions
- 11. Arrays
  - a. One-dimensional arrays
  - b. Multi-dimensional arrays
  - c. Null terminated strings (as arrays)
  - d. Insert, delete, and sequential search for unordered list
  - e. Insert, delete, and binary search for ordered list
- 12. Enumeration types

## Assignment:

1. Maintain a reading schedule for the text(s), approximately 30 pages per week.
2. Write programs using the C++ programming language.
3. Test and debug programs.
4. Write program documentation.
5. Take objective examinations including final exam.

## 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, hands-on 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:

Programming and Problem Solving with C++, by Nell Dale - Jones and Bartlett, 5th edition, 2010