# 8/15/2020

# brcc keystone logo

Baton Rouge Community College

*Academic Affairs Master Syllabus*

Date Approved: 2 September 2020

Term and Year of Implementation: Fall 2020

**Course Title:** Introduction to Data Structures and Algorithms

**BRCC Course Rubric:** CSCI 2103

**Previous Course Rubric**: CSCI 210

**Lecture Hours per week-Lab Hours per week-Credit Hours**: 3-0-3

**Per semester: Lecture Hours-Lab Hours-Instructional Contact Hours**: 45-0-45

**Louisiana Common Course Number:**

**CIP Code:** 11.0701

**Course Description:** Presents the related theory for representing and accessing information using a higher level programming language. Studies concepts of data types, data abstraction, data structures, and advanced programming techniques.

**Prerequisites:** CSCI 1943 (or CSCI 194) and MATH 1113 (or MATH 101) or MATH 1213 (or MATH 110)

**Co-requisites:** None

**Suggested Enrollment Cap:** 25

**Learning Outcomes.** *Upon successful completion of this course, the students will be able to:*

1. Develop abstract data types, classes, and objects using C++.

2. Describe data storage techniques.

3. Describe the basic concpts of design and analysis with an emphasis on searching and sorting algorithms.

4. Apply basic data structures.

5. Write C++ program to solve problems.

**Assessment Measures.** Assessment of all learning outcomes will be measured using the following methods:

1. Student participation

2. Programming assignments

3. Instructor-prepared tests

4. Formal presentations

5. Homework assignments

**Information to be included on the Instructor’s Course Syllabi:**

* ***Disability Statement*:** Baton Rouge Community College seeks to meet the needs of its students in many ways. See the Office of Disability Services to receive suggestions for disability statements that should be included in each syllabus.
* ***Grading:*** The College grading policy should be included in the course syllabus. Any special practices should also go here. This should include the instructor’s and/or the department’s policy for make-up work. For example in a speech course, “Speeches not given on due date will receive no grade higher than a sixty” or “Make-up work will not be accepted after the last day of class”.
* ***Attendance Policy*:** Include the overall attendance policy of the college. Instructors may want to add additional information in individual syllabi to meet the needs of their courses.
* ***General Policies*:** Instructors’ policy on the use of things such as beepers and cell phones and/or hand held programmable calculators should be covered in this section.
* ***Cheating and Plagiarism*:** This must be included in all syllabi and should include the penalties for incidents in a given class. Students should have a clear idea of what constitutes cheating in a given course.
* ***Safety Concerns:*** In some courses, this may be a major issue. For example, “No student will be allowed in the lab without safety glasses”. General statements such as, “Items that may be harmful to one’s self or others should not be brought to class”.
* ***Library/ Learning Resources:*** Since the development of the total person is part of our mission, assignments in the library and/or the Learning Resources Center should be included to assist students in enhancing skills and in using resources. Students should be encouraged to use the library for reading enjoyment as part of lifelong learning.

**Expanded Course Outline:**

I. An Overview of Computers and Programming Languages

A. Introduction

B. A brief overview of the History of Computers

C. Elements of a Computer System

1. Hardware

2. Software

D. The Language of a Computer

E. The Evolution of Programming Languages

F. Processing a High-Level Language Program

G. Programming with the Problem Analysis-Coding-Execution Cycle

H. Object-Oriented Programming

I. ANSI/ISO Standard C++

II. Basic Elements of C++

A. The Basics of a C++ Program

B. Data Types

1. Simple Data Types

2. Floating-Point Data Types

3. The String Type

C. Arithmetic Operators and Operator Precedence

1. Order of Precedence

D. Expressions

1. Mixed Expressions

2. Type Conversion (Casting)

E. Input

1. Allocating Memory with Constants and Variables

2. Putting Data into Variables

F. Increment and Decrement Operators

G. Output

H. Preprocessor Directives

1. Using Cin and Cout in a Program and namespace

2. Using the string Data Type in a Program

I. Program Style and Form

1. The Function Main

2. Syntax

3. Use of Blanks

4. Use of Semicolons, Brackets, and Commas

5. Semantics

6. Form and Style

7. Documentation

J. More on Assignment Statements

III. Input/Output

A. I/O Streams and Standard I/O Devices

1. Cin and the Extraction Operator

B. Using Predefined Functions in a Program

1. Cin and the get Function

2. Cin and the ignore Function

3. The Putback and peek Functions

4. The Dot Notation Between I/O Stream Variables and I/O

5. Functions: A Precaution

C. Input Failure

1. The Clear Function

D. Output and Formatting Output

1. Setprecision

2. Fixed

3. Showpoint

4. Setw

E. Additional Output Formatting Tools

1. Fill and setfill

2. The left and right Manipulators

3. The Flush Function

4. Input/Output and string Type

F. File Input/Output

IV. Control Structures I (Selection)

A. Control Structures

B. Relational Operators

1. Relational Operators and Simple Data Types

2. Relational Operators and the string type

C. Logical (Boolean) Operators and Logical Expressions

1. Order of Precedence

2. Short-Circuit Evaluation

3. The int Data Type and Logical (Boolean) Expressions

4. The bool Data Type and Logical (Boolean) Expressions

D. Selection: if and if..else

1. One-way Selection

2. Two-way Selection

3. Compound (Block of) Statements

4. Multiple Selections: Nested if

5. Comparing if…else Statements with a series of if Statements

6. Using Pseudocode to Develop, Test, and Debug a Program

7. Input Failure and the if Statement

8. Confusion Between the Equality Operator (==) and the Assignment Operator (=)

9. The Conditional Operator (?:)

10. Switch Structures

11. Terminating a Program with the assert Function

V. Control Structures II (Repetition)

A. Why is Repetition needed?

B. The While Looping (Repetition) Structure

1. Case 1: Counter-Controlled while Loops

2. Case 2: Sentinel-Controlled while Loops

3. Case 3: Flag-Controlled while Loops

4. Case 4: EOF-Controlled while Loops

C. The eof Function

D. The for Looping (Repetition) Structure

E. The do…while Looping (Repetition) Structure

F. Break and Continue Statements

G. Nested Control Structures