Computer Science – CSCI 110 -

INTRODUCTION TO COMPUTER SCIENCE I
Fall 2017 Course Syllabus
Crafton Hills College

Basic Course Information

Meeting Time:	MoWe 7:00 –7:50 pm (Lecture) MoWe 8:00 – 9:15 pm (Lab)
Duration:	August 14 – December 15, 2017 (18 weeks)
Location:	CENT 121 & online
Prerequisites:	MATH 102: College Algebra or MATH 160: Precalculus or eligibility for MATH 250
Credit Units:	4

Instructor Information

Instructor:	Anas “David” Nimri
Email:	animri@sbccd.cc.ca.us
Office:	CENT 135
Office Hours: **(CENT 135)**	We 9:15 – 9:45 pm and by appointments

Course Description

Introduction to computer science, including programming, theory, and applications. Introduction to computer software design, implementation, methods, environment, and problem solving using structured programming in a high-level language. Topics include data types, variables, expressions, input/output, control structures, functions, arrays, strings, and introduction to object-oriented programming.

Student Learning Outcomes

1. Student will analyze problems and design algorithms in pseudocode and in a high-level language.
2. Student will read, understand, and trace the execution of programs written in the C++ language.
3. Student will design and test C++ programs using appropriate and valid variables, expressions, selection statements, loops, functions, and arrays.
4. Student will be able to describe the software development life cycle.

Program-Level Student Learning Outcomes

1. Students will demonstrate the ability to apply analytical and logical thinking to gathering and analyzing information, designing and testing solutions to problems, and formulating plans.
2. Students will demonstrate the ability to work collaboratively in a team setting.

Course Objectives

Upon satisfactory completion of the course, students will be able to:

1. Analyze and explain the behavior of programs involving fundamental programming constructs.
2. Describe the principles of structured programming.
3. Distinguish between structured programming and object-oriented programming.
4. Design, implement, test, and debug structured programs using fundamental programming constructs: basic computation, simple I/O, standard conditional and iterative control structures, and functions.
5. Choose appropriate conditional and iterative control structures for a given programming task.
6. Apply the techniques of structured decomposition to break a program into smaller pieces.
7. Describe the mechanics of parameter passing.
8. Describe the software development life cycle.
9. Discuss the importance of algorithms in the problem-solving process.
10. Create algorithms for solving problems.

11. Use pseudocode or a programming language to implement, test, and debug algorithms for solving problems.
12. Utilize programming tools and environments to develop and test programs.

Required Textbooks

Programming in C++ with Data Structures, zyante.com

1. 1. Sign up at **zyBooks.com**
2. Enter zyBook code: **CRAFTONHILLSCSCI110NimriFall2017**
3. Click 'Subscribe'
4. Cost: $65

Other Materials/Supplies

Flash drive (at least 4GB recommended) or other portable / cloud-based storage for lab assignments.
Note pad/blank paper and pen/pencil.

Blackboard Course Website

A Blackboard online site has been set up for this course. Assignments and other important course materials will be available on this site. You may access Blackboard through the Crafton Hills College website (http://www.craftonhills.edu, under *Logins* on the right hand side of the home page) or directly at http://sbccd.blackboard.com. Your username will be your student ID number (not your name). Your password is also your student ID number (unless you have changed it).

Computer Lab – CENT 135 - (909)389-3374

The CIS Computer Lab is located in the Central building room 135. Open lab hours are posted on the outside of the door. The lab is available for students to work on their computer assignments, and help is available from lab tutors/techs. While lab tutors/techs will not do your work for you, they may help you understand the assignments and the steps necessary to complete them.

Assessment Methods

Your course grade will be based on the following evaluation categories (the percentages are approximated):

Evaluation	**%**	**Description**
Homework Exercises	10	Exercises will cover the concepts taught in the course.
Laboratory Exercises	20	Graded hands-on lab activities will be completed during class time. If a student misses that day they will receive a 0 for the activity. Lab activities **CANNOT** be made up.
Programming Projects	20	Students will work on a number of individual programming projects as well as a group project. Projects involve design, implementation, and documentation of larger-scaled programs.
Examinations and Tests	50	There will be two Test and two exams—midterm and final—given during the semester. All test and exams will be completed in class. Unless otherwise specified.
Extra Credit	10	Participation & “pop quizzes.” Pop quizzes are extra credit (so they do not negatively impact your grade) but are designed to encourage students to read ahead, and thus, pop quizzes will cover the weekly course material not yet discussed in class.

Grading Scale

Your final course grade will be assigned based on the following scale:

A	90.00% or above
B	80.00% — 89.99%
C	70.00% — 79.99%
D	60.00% — 69.99%
F	0% — 59.99%

Your recorded grades will be available for your review on Blackboard. Please maintain copies of your assignments and keep all graded work. Notify the instructor immediately if you find any discrepancies.

Policies on Communication, E-mail, and Technical Issues

All students are required to use their campus assigned email. Any emails sent to the instructor with questions, concerns, or issues will be answered within 2 business days. **The subject line of the email must include the course number and your first and last name. Emails without subject line information will *not* be read.** Students may also contact the instructor at the posted phone numbers. Please do not call after 10 PM California time. If you leave a message, please leave your first and last name and a return phone number. Calls will be returned within 1-2 business days.

The Announcements area of Blackboard will be used to post daily reminders, changes, revisions or issues related to the course. Be sure to check daily. **Email announcements will be sent to your official student email address given by the school district SBCCD.**

The instructor will monitor the Blackboard site. If the Blackboard site experiences technical difficulties that impact assignments, tests or exams, due dates may be adjusted by instructor. If you run into a technical problem that impacts your ability to complete your work on time, you must contact the instructor well *before* the due date to discuss the problem.

Policies on Assignment Submission

When submitting your source code or documents, please submit using the following name scheme (with underscores not dashes):

Labs: labX_LastNameFirstName.cpp or labX_LastNameFirstName.doc
Homework: hWX_LastNameFirstName.cpp or hWX_LastNameFirstName.doc
Project: projX_LastNameFirstName.cpp or projX_LastNameFirstName.doc

Here is an example: example: lab5_NimriDavid.cpp
These names will make it much easier to run your source code to test, saving me time, allowing me to grade your assignments faster.

Attendance Policy

You are expected to attend and be on time to every class meeting. Attendance will be taken at each class meeting. Students are responsible for dropping the course officially if they no longer wish to be enrolled. Failure to do so may result in a failing grade and/or a financial obligation to the college. At my discretion, I may drop or fail a student from the course if the student is absent for **three** or more class meetings unless prior arrangements have been made with the instructor.

Late Work/Make-up Policy

You are responsible for starting your work early and completing them by their deadlines. **No late work will be accepted for full credits** except under documented extreme emergency situations that are communicated to the instructor *prior* to the due date.

Each student is allowed to make up **at most *two* assignments within one week after their deadlines with at least 10% grade deduction**. For example, if an assignment is worth 100 points, the late assignment can receive at most 85 points. Students should notify the instructor immediately if they intend to make up an assignment (the makeup assignment may or may not be the same as the original). The two make-up allowances are included to provide some flexibility in accommodating students who may need additional time to complete assigned work; however, students are highly discouraged to submit late work unless it is absolutely necessary. Avoid falling behind in your coursework by following the course schedule closely and seeking help as needed.

Make-up exams will only be given to students who make prior arrangements with the instructor to take the exam *before* the scheduled exam day. No make-up exams can be scheduled after the class has already taken the exam.

Classroom Rules

- ✓ No cell phones or texting are allowed during class. If you have an emergency situation that requires accepting incoming phone calls or text, please notify the instructor in advance.
- ✓ No food or drinks are allowed in the classroom.
- ✓ Do not use computers unless they are used as parts of lab or class exercises during class.

Expectations, Goals, and Advice

Assignments will be assigned at least one week before their due dates. Assignments, lectures, and other important course materials will be posted on Blackboard in a timely manner.

Faculty has a responsibility to maintain a positive learning environment for all students. Students share in this responsibility. In order to get the most benefit from and succeed in this class, you should:

- ✓ Attend all classes, be on-time, and participate fully in class activities. Any work done in-class cannot be made up. If you must miss a class, be sure to check with a fellow student for any announcements or changes to the course materials.
- ✓ Give yourself ample time to complete assignments outside of class. This is a college-level course and as such there are numerous assignments, exercises, and reading to be completed outside of our regular class time.
- ✓ Complete your reading assignments *before* the material is discussed. This way the terms in the lecture will be familiar and you will know what questions you want to ask. Extra credit quizzes will be given to encourage you to read.
- ✓ **Ask questions!** Schedule conferences with me when help is needed. Come to CIS computer lab and work with the lab tutors on projects if you need additional help.
- ✓ Buddy up with another student or students. Exchange phone numbers and/or email addresses, share notes, discuss assignments. You are encouraged to work together, discuss class materials, assignments and projects and to help each other as individual interests and abilities permit. However, each student is responsible for his/her own work. **Copying another student's answers or using another student's project files is considered cheating and will result in a zero on the project or exam for *all* students involved, who in addition may be subject to the College's disciplinary actions.**
- ✓ Complete all assignments on time. Each homework, project or exercise is designed to prepare you for the next. If you are having difficulties come see me. **Do not wait until the work is late**!

Academic Honesty

Plagiarism and cheating will not be tolerated. Students who commit plagiarism or cheating will receive a zero on the assignment or exam and will also be subject to the College's disciplinary actions which may include the student grade set to an "F" for the entire course.

Accommodations for Students with Disabilities

Students with a disability who may need special accommodation(s) for this class are encouraged to notify the instructor and contact **Disabled Student Programs & Services** (909)389-3325 as soon as possible.

Important Dates

Please review the campus calendar:
http://www.sbccd.org/~/media/Files/SBCCD/District/Business%20Services%20Documents/academic-calendar/2016-2017-acad-calendar-6-18-15.pdf

Overview of Course Topics and Tentative Schedule

Week	Week Of	Lecture Topic	Reading	Assignment
1	**8/14**	Introduction to computer science Programming and Problem-Solving -Defining algorithms -Problem solving strategies -Program design -Writing pseudocode **Lab 1 Algorithms**		
		1.9 Computers and programs 1.10 A brief tour of a computer	Chapter 1 Introduction	Chapter 1 Zyante

		1.11 Language history Survey of current languages **Lab 2 Program design tools and programming environments**		
2	**8/21**	Binary notation & math Day **Lab 3 Software development life cycle**	Numbering Systems (not in the book)	
		1.2 A first program 1.3 Basic output 1.4 Basic input 1.5 Comments and whitespace 1.6 Errors/Warnings **Lab 4 Calculations**	Chapter 9 Streams	
3	**8/28**	2.1 Variables (int) 2.2 assignments 2.3 Identifiers 2.4 Numbers in binary 2.5 Expressions (int) 2.6 Floating-point data types Project Work Day	Chapter 2 Variables/ Assignments	*Chapter 1 Zyante due* Chapter 2 Zyante Homework 1 Project 1
		2.7 Const vars 2.8 Math functions 2.9 Type conversions 2.10 Int overflow 2.11 Numeric types 2.12 Unsigned 2.14 Character data type 2.15 String 2.16 Random number generation 2.17 Debugging 2.18 Style guide 9.1 ostream/cout 9.2 istream/cin 9.3 output formatting 9.7 extraction before getline **Lab 5 Debugging and Health Data Calculator**		
4	**9/4**	**Campus is closed**		
		3.1 If-else statement 3.2 Relational and equality operators 3.3 Multiple if-else **Lab 6 If-else**	Chapter 3 Branches	*Chapter 2 Zyante due* Chapter 3 Zyante
5	**9/11**	3.4 Logical operators 3.5 Switch statement 3.6 Boolean data type Additional Lab/project time		
		Review		*Homework 1 due* Homework 2
6	**9/18**	**TEST 1 (ch. 1-2)**		
		3.7 String compare 3.8 String access ops 3.9 String modify ops 3.10 Char ops		*Project 1 due* Project 2 *Chapter 3 Zyante due*

		3.11 Conditional expressions 3.12 Floating-point comparison **Lab 7 More Branching**		
7	**9/25**	4.1 Loops 4.2 While loops 4.3 More while examples 4.4 Counting **Lab 8 String Manipulation**	Chapter 4 Loops	Chapter 4 Zyante
		4.5 For loops 4.6 Nested loops 4.7 incremental programming **Lab 9 While Loops**		
8	**10/2**	4.8 Break/continue 4.9 Do-while loops 4.10 Enumerations **Lab 10 Dice Statistics**		
		Review Lab – nested loop (extra credit)		*Chapter 4 Zyante due* *Homework 2 due*
9	**10/9**	**Midterm (chapters 1 – 4)**		Chapter 5 Zyante
		5.1 Array concept 5.2 Arrays 5.3Vectors 5.4 Array/vector iteration drill 5.5 Iterating through vectors **Lab 11 Arrays**	Chapter 5 Arrays/Vectors	Homework 3
10	**10/16**	5.6 Vector resize 5.7 Vector push_back 5.8 Multiple vectors 5.6 Vector resize 5.7 Vector push_back 5.9 Modifying/copying a vector 5.10 Arrays vs. vectors **Lab 12 Vectors**		*Project 2 due* Project 3
		5.11 Swap 5.13 Two-dimensional arrays 5.14 Engineering examples 5.15 Char arrays -- C strings 5.16 String library functions 5.17 Char library functions: ctype **Lab 13 2D Array**		
11	**10/23**	Search and sorting algorithms **Lab 14**	Chapter 6 User-Defined Functions	*Chapter 5 Zyante due* Chapter 6 Zyante
		6.1 User-defined function basics 6.2 Function parameters 6.3 Return value 6.4 Reasons for defining functions Project Work Day		
12	**10/30**	6.5 Function with branches/loops 6.6 Unit testing 6.7 How functions work 6.8 Functions: Common errors **Lab 15 Functions**		
		6.9 Passing arguments by reference 6.10 String/vector parameters 6.11 Scope of variable/function definitions 6.12 Default parameter values		Project 3 due Project 4

		Project Work Day		
13	**11/6**	6.13 Function name overloading 6.14 Engineering examples 6.15 Command-line arguments 6.16 Preprocessor 6.17 Separate files 6.18 Modular compilation **Lab 16 Functions & Testing**		*Chapter 6 Zyante due*
		9.4 String streams 9.5 File input/output 9.6 Command-line Arguments and Files	Chapter 9 Streams	*Homework 3 due*
14	**11/13**	**Project Work Day**		
		Review		
15	**11/20**	**TEST 2 (ch. 5-6 & 9)**	Chapter 7 Objects/Classes (7.1 – 7.11)	Homework 4 *Chapter 7 Zyante (7.1 – 7.11)*
		7.1 – 7.4 Structs: Grouping data Structs and functions Structs and vectors Structs, vectors, and functions: A seat reservation example Project Work Day		
16	**11/27**	7.5 Objects: Intro 7.6 Classes: Intro 7.7 Classes/vectors/functions example: Seat reservations 7.8 Mutators, accessors, and private helpers **Lab 17 OOP**		
17	**12/4**	7.9 Class constructors 7.10 Constructor overloading 7.11 Constructors: Member initialization **Lab 18 OOP2 (continuation of lab 17)**		
		12.1 Exception basics 12.2 Exceptions with functions 12.3 Multiple handlers Review	Chapter 12 Exceptions	*Homework 4 due* *Chapter 7 Zyante due* Final Project due
18	**12/11**	**Final Exam (cumulative)***		
	12/13	**Alternative Final Exam Day (cumulative)***		

***Final exam date to be determined by instructor.**

This syllabus & dates are subject to change at any time.

I look forward to working with you!

"Do not worry about your difficulties in Mathematics. I can assure you mine are still greater." – Albert Einstein