El Camino College

COURSE OUTLINE OF RECORD - Official

I. GENERAL COURSE INFORMATION

Subject and Number: Computer Science 7
Descriptive Title: The Beauty of Computer Science Principles

Course Disciplines: Computer Science

Division: Mathematical Sciences

Catalog Description: This course introduces students to the central ideas of computer science, inviting students to develop the computational thinking vital for success in computer programming and across multiple disciplines. The course is unique in its focus on fostering students to be creative and encouraging students to apply innovative processes when developing computational artifacts such as avisualization, graphic or movie. Students will design and program inventive solutions to computational problems using an iterative process similar to what computer scientists, engineers, artists, and writers use to bring ideas to life.

Conditions of Enrollment: **Prerequisite**

credit in
Mathematics 37 or

Mathematics 40
with a minimum grade of C in prerequisite
or

qualification by testing (El Camino College Mathematics Placement Test) and assessment

Course Length: [X] **Full Term** [ ] **Other (Specify number of weeks):**
Hours Lecture: **3.00 hours per week** [ ] **TBA**
Hours Laboratory: **3.00 hours per week** [ ] **TBA**
Course Units: **4.00**

Grading Method: **Letter**
Credit Status **Associate Degree Credit**

Transfer CSU: [X] **Effective Date: Proposed**
Transfer UC: [X] **Effective Date: Proposed**

General Education:
El Camino College: **4B – Language and Rationality – Communication and Analytical Thinking**

Term: Other:

CSU GE:

IGETC:

II. OUTCOMES AND OBJECTIVES

A. COURSE STUDENT LEARNING OUTCOMES (The course student learning outcomes are listed below, along with a representative assessment method for each. Student learning outcomes are not subject to review, revision or approval by the College Curriculum Committee)

1. Write Proper algorithms. (Properly analyze a problem using top down design, and write an algorithm that can be translated into computer code)
2. Analyze technological innovations with respect to their beneficial and harmful impact on society
3. Use the idea of abstraction to reduce the complexity of problems.

The above SLOs were the most recent available SLOs at the time of course review. For the most current SLO statements, visit the El Camino College SLO webpage at http://www.elcamino.edu/academics/slo/.

B. Course Student Learning Objectives (The major learning objective for students enrolled in this course are listed below, along with a representative assessment method for each)

1. Develop an algorithm for implementation in a program.

   Other (specify)

   Programming Assignment
2. Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge

   Other (specify)

   Computer Programming Assignment
3. Use functions to manage complexity in programs.

   Other (specify)

   Programming Assignment
4. Create a computational artifact for creative expression.

   Other (specify)

   Computational Artifact (a visualization, graphic or video)
5. Analyze how data representations, storage, security, and transmission of data involve computational manipulation of information.

   Term or other papers
6. Explain the connections between computing and real-world contexts, including economic, social, and cultural contexts.

   Term or other papers
7. Explain the abstractions in the Internet and how the Internet functions.

   Objective Exams

III. OUTLINE OF SUBJECT MATTER (Topics are detailed enough to enable a qualified instructor to determine the major areas that should be covered as well as ensure consistency from instructor to instructor and semester to semester.)

Lecture or Lab	Approximate Hours	Topic Number	Major Topic
Lecture	7	I	Abstraction A. Detail Removal B. Generalization C. Base Conversions D. Digital Data E. Functions
Lecture	2	II	Lists and Scoping A. Variables B. Scope
Lecture	9	III	Algorithms A. Historical Algorithms B. Properties of Algorithms C. Algorithms vs. Functions & Procedures D. Different Programming Languages
Lecture	2	IV	Algorithmic Complexity A. Running Time Analysis B. Algorithm Correctness
Lecture	5	V	Programming Paradigms A. Functional B. Imperative C. Object Oriented D. Declarative
Lecture	4	VI	Concurrency A. Processor Overview B. Moore's Law C. Threads D. Non Deterministic Code E. Deadlocks
Lecture	2	VII	Artificial Intelligence A. History B. Natural Language Processing C. Vision (Perception) D. Neural Networks E. Robotics F. Superintelligence
Lecture	3	VIII	Recursion A. Finding Subproblems

			B. Forming recursive calls C. Tracing recursion
Lecture	5	IX	Social Implications of Computing A. Privacy Issues B. Social Movement C. Job Implications D. Copyright
Lecture	2	X	Data A. Lossless vs Lossy B. Metadata C. Correlation and causality D. Visualization
Lecture	6	XI	Python A. Basics B. Control Flow C. Operations on Lists
Lecture	5	XII	The Internet A. Autonomous Systems B. IP Addresses C. The World Wide Web D. Web 2.0 E. Packet Switching F. Bandwidth Latency G. Cryptography
Lecture	2	XIII	Limits of Computing A. Tractability B. Solvability
Lab	6	XIV	Abstraction using *SNAP!* A. Detail Removal B. Generalization C. Base Conversions D. Digital Data E. Functions
Lab	3	XV	Lists and Scoping using *SNAP!* A. Variables B. Scope
Lab	3	XVI	Algorithms using *SNAP!* A. Historical Algorithms B. Properties of Algorithms C. Algorithms vs. Functions & Procedures D. Different Programming Languages

Lab	3	XVII	Algorithmic Complexity using *SNAP!* A. Running Time Analysis B. Algorithm Correctness
Lab	3	XVIII	Concurrency using *SNAP!* A. Processor Overview B. Moore's Law C. Threads D. Non Deterministic Code E. Deadlocks
Lab	3	XIX	Recursion using *SNAP!* A. Factorial Problem B. Fibonacci Problem
Lab	3	XX	Data using *SNAP!* A. Lossless vs Lossy B. Metadata C. Correlation and causality D. Visualization
Lab	6	XXI	Python A. Basics B. Control Flow C. Operations on Lists
Lab	4	XXII	The Internet A. Autonomous Systems B. IP Addresses C. The World Wide Web D. Web 2.0 E. Packet Switching F. Bandwidth Latency G. Cryptography
Lab	8	XXIII	Explore Performance Task A. Selection of computing innovation B. Analysis of Social, Economic, and Cultural Impact C. Creation of Visual Artifact D. Written report
Lab	12	XXIV	Create Performance Task A. Inclusion of basic programming elements B. Use of Abstraction C. Demonstrate the use of algorithms D. Collaborative Reflection on Programs and Process E. Individual Reflection on Programs and Process

Total Lecture Hours	54
Total Laboratory Hours	54
Total Hours	108

IV. PRIMARY METHOD OF EVALUATION AND SAMPLE ASSIGNMENTS

A. PRIMARY METHOD OF EVALUATION:

Problem solving demonstrations (computational or non-computational)

B. TYPICAL ASSIGNMENT USING PRIMARY METHOD OF EVALUATION:

Description

The goal of this homework assignment is to create a word guessing game between two human players. Player 1 provides a word and Player 2 tries to guess what it is using feedback she received from previous guesses.

Player 1 first enters the secret word by entering a response to an "ask" block. The word is *immediately hidden* from view after Player 1 finishes entering it. The computer then starts by telling Player 2 how many letters are in the mystery word. After Player 2 makes a guess, the computer should tell the player whether her guess was right or wrong. Remember guesses should be case-insensitive. If Player 2 was wrong, the computer should also tell Player 2 how many letters in her guess were correctly positioned in the actual answer. The computer does not, however, have to say which letters were correct or which positions they were in. The conversation alternates back and forth until Player 2 guesses the secret word correctly.

C. COLLEGE-LEVEL CRITICAL THINKING ASSIGNMENTS:

1. Performance Task: Explore — Impact of Computing Innovations

   Computing innovations have had considerable impact on the social, economic and cultural areas of our lives. To focus your work on this task, select a computing innovation that has significant impact, or the potential for significant impact on our society, economy, or culture. Also include the potential for both beneficial and harmful effects.

   A. General Requirements

   For this performance task, you are required to choose an innovation that has a significant effect on some population. The effect could be a small effect on more than a hundred people, or a very large effect on a smaller number of people.

   B. Written Requirements

You will write responses to specific prompts associated with content requirements. Your responses should convey a deep level of understanding about your innovation and its impacts. Your responses must also include information learned from your references.

C. Visual Artifact

Choose one of the potential beneficial or harmful effects of the innovation you described in your previous response and use a computer to create a visual artifact (a visualization, graphic, or movie) related to it.

D. References

Include at least two – and no more than five – references/citations to sources used to formulate your responses to this performance task.

2. Performance Task: Create — Creative Programming

Programming is a creative process that involves individual and collaborative effort to bring ideas to life through designing, developing, testing, and debugging programs.

For this task, you and your partner will individually and collaboratively develop programs of your choosing. You will be asked to answer questions about your programs, and you will be asked to provide details about how you and your partner collaborated.

A. General Requirements

For this performance task, you are required to work as a team to develop a program together and prepare a rationale about it. Each team member should create at least one significant part of the program.

B. Program Requirements

You will develop one program with your partner and one program on your own. The program you produce individually must be different from the one you write collaboratively and from your partner's individually produced program. You could choose to add additional features to the collaboratively-produced program, or start a new program. The program you write independently can be written in the same language or in a different language as the program that you write collaboratively.

D. **OTHER TYPICAL ASSESSMENT AND EVALUATION METHODS:**

Other exams

Written homework

Homework Problems

Term or other papers

Other (specify):

Programming Assignments

V. INSTRUCTIONAL METHODS

Laboratory

Lecture

Note: In compliance with Board Policies 1600 and 3410, Title 5 California Code of Regulations, the Rehabilitation Act of 1973, and Sections 504 and 508 of the Americans with Disabilities Act, instruction delivery shall provide access, full inclusion, and effective communication for students with disabilities.

VI. WORK OUTSIDE OF CLASS

Study

Required reading

Problem solving activities

Written work

Estimated Independent Study Hours per Week: 6

VII. TEXTS AND MATERIALS

A. **UP-TO-DATE REPRESENTATIVE TEXTBOOKS**
G. Michael Schneider; Judith Gersting. Invitation to Computer Science. 7th ed. National Geographic/Cengage Learning, 2016.

B. **ALTERNATIVE TEXTBOOKS**

C. **REQUIRED SUPPLEMENTARY READINGS**
Blown to Bits: Your Life, Liberty, and Happiness After the Digital Explosion, (http://www.bitsbook.com/), 2008

Discipline Standard

D. **OTHER REQUIRED MATERIALS**

VIII. CONDITIONS OF ENROLLMENT

A. Requisites (Course and Non-Course Prerequisites and Corequisites)

Requisites	Category and Justification
Course Prerequisite Mathematics-37 or	Computational/Communication Skills
Course Prerequisite Mathematics-40 or	Computational/Communication Skills
Non-Course Prerequisite	This is a standard phrase to allow for student prerequisite challenges.

B. Requisite Skills

Requisite Skills
Utilize variables in expressions, tables and graphs. MATH 40 - Starting with a linear model in tabular, graphical or symbolic form, translate the model into the other two forms. MATH 37 - Recognize and apply the concepts of variable, expression, equation and function.
Plot points on a cartesian plane. MATH 40 - Graph linear equations and systems of linear equations by plotting points or by using intercepts and the slope. MATH 37 - Set up, graph, and solve linear equations, systems of linear equations, and linear inequalities using a variety of techniques.

C. Recommended Preparations (Course and Non-Course)

Recommended Preparation	Category and Justification

D. Recommended Skills

Recommended Skills

E. Enrollment Limitations

Enrollment Limitations and Category	Enrollment Limitations Impact

Course created by Russell Solomon on 10/21/2016.

BOARD APPROVAL DATE: 01/16/2018

LAST BOARD APPROVAL DATE:

Last Reviewed and/or Revised by Solomon Russell on 10/21/2016

19763