California State University, San Marcos General Education Program
GENERAL EDUCATION NEW COURSE CERTIFICATION REQUEST
• AREA B4: Mathematics and Quantitative Reasoning

See GE Handbook for information on each section of this form

ABSTRACT

Course Abbreviation and Number: CS 105	**Course Title: Media-propelled Computational Thinking**
Number of Units:3 _____	

College or Program: CHABSS X CSM CEHHS COBA Other_______________________________	**Desired term of implementation:** X Fall Spring Summer Year2014
Course Proposer (please print):Rocio Guillen	**Email:rguillen@csusm.edu**

1. Course Catalog Description: A media-propelled introduction to computation. Programming languages uch as Alice, Java, Python, or Jython are studied and programming techniques are used to examine first the basic functions that draw objects, including lines and curves, and later to explore familiar physical, biological, or other scientific processes. Mathematical competence necessary for academic success will be enhanced. May not be taken for credit by students who have received credit for CS 200-2. Enrollment Requirement: Completion of the Entry-Level Mathematics (ELM) requirement.

2. GE Syllabus Checklist: The syllabi for all courses certified for GE credit must contain the following:

	Course description, course title and course number
	Student learning outcomes for General Education Area and student learning objectives specific to your course, linked to how students will meet these objectives through course activities/experiences
	Topics or subjects covered in the course
	Registration conditions
	Specifics relating to how assignments meet the writing requirement
	Tentative course schedule including readings
	Grading components including relative weight of assignments

SIGNATURES

____
Course Proposer Date

____
Department Chair date

Please note that the department will be required to report assessment data to the GEC annually. ______
DC Initial

_____________________ Support ☐ Do not support* ☐
Library Faculty Date

_____________________ Support ☐ Do not support* ☐
Impacted Discipline Chair Date

_____________________ Support ☐ Do not Support* ☐
Impacted Discipline Chair Date

_____________________ Approve ☐ Do not Approve ☐
GEC Chair Date

California State University, San Marcos General Education Program
GENERAL EDUCATION NEW COURSE CERTIFICATION REQUEST
• AREA B4: Mathematics and Quantitative Reasoning
See GE Handbook for information on each section of this form
*** If the proposal is not supported, a memo describing the nature of the objection must be provided.**

Course Coordinator: Phone Email:

California State University, San Marcos General Education Program
GENERAL EDUCATION NEW COURSE CERTIFICATION REQUEST
• AREA B4: Mathematics and Quantitative Reasoning
See GE Handbook for information on each section of this form

Part A: B4 Quantitative Reasoning General Education Learning Outcomes (GELOs) related to course content. [Please type responses into the tables.]

Math/Quant Reasoning GELOs this course will address:	Course content that addresses each GELO.	How will these GELOs be assessed?
B4.1: Explain and apply a variety of fundamental mathematical concepts, symbols, computations and principles.	Concepts: Numeric bases. Converting from one numeric base to another numeric base . Boolean algebra used in programming constructs. Analytic and geometry knowledge to think and reason in terms of 3-D. Classes and objects to implement graphics and sound. Basic constructs to write a program that is syntactically and semantically correct: decision statements, iteration statements, functions. Symbols: flowcharts, scripts, story boards, arithmetic operators: binary +, unary -, binary -, binary *, binary /, binary %, logical operators: &&, ||, !; relational operators: >, >=, <, <=, ==, !=; assignment operators : =, +=, -=, *=, /=, %=. Computations: programs are written, compiled, debugged and tested for correct application of constructs. Principles: planning a solution for a given problem, designing the solution using a flowchart, script or story board to review the logic of the solution, translating the solution into code following the syntax of a programming language, compiling the code to remove any syntactic errors, testing the program with several test cases to correct logic errors, documenting code for ease of maintenance.	Students will be given a problem involving one or more of the concepts learned and be expected to solve it using the relevant symbols, computations and principles. Students will be expected to state their solution in a correct, clear and complete manner.
B4.2: Determine which quantitative or symbolic reasoning methods are appropriate for solving a given problem and correctly implement those methods.	Given a problem plan a solution, design the solution using a flowchart. Follow the syntax of the statements in a programming language to write a program.	The student will be given a problem whose solution will require the design and implementation of a program. The student will have to make choices about which classes and objects to use and how to put them together for the program to generate the correct results.

California State University, San Marcos General Education Program
GENERAL EDUCATION NEW COURSE CERTIFICATION REQUEST
• AREA B4: Mathematics and Quantitative Reasoning

See GE Handbook for information on each section of this form

Part B: General Education Learning Outcomes required of all GE courses related to course content:

GE Outcomes required of all Courses	Course content that addresses each GE outcome?	How will these GELOs be assessed?
Students will communicate effectively in writing to various audiences. (writing)	Assignments, quizzes, exams and laboratory work will require students to describe solutions to the problem given, design the solution using pseudocode or a flowchart, translate their solution to code using a programming language, and document the program for other users to enhance and maintain.	Students will be expected to write out solutions to problems showing the logic and flow of information of their solution, coding the solution using a programming language, and documenting the program in detail so that other users, in particular other students, read and understand the program for further development.
Students will think critically and analytically about an issue, idea or problem. (critical thinking)	Problems given will require students to determine what is being asked of them, think about what method/procedure of solution is appropriate, and properly implement that method/procedure.	Students will be expected to solve problems by designing the solution using pseudocode or a flowchart, translating their solution to code using a programming language, and documenting the program for other users to enhance and maintain.

Part C: GE Programmatic Goals: The GE program aligns with CSUSM specific and LEAP Goals. All B4 courses must meet at least one of the LEAP Goals.

GE Programmatic Goals	*Course addresses this LEAP Goal:*
LEAP 1: Knowledge of Human Cultures and the Physical and Natural World.	***No Yes*** ✓
LEAP 2: Intellectual and Practical Skills	***No Yes*** ✓
LEAP 3: Personal and Social Responsibility	***No Yes*** ✓
LEAP 4: Integrative Learning	***No Yes*** ✓
CSUSM Specific Programmatic Goals	**Course content that addresses the following CSUSM goals. Please explain, *if applicable.***
CSUSM 1: Exposure to and critical thinking about issues of diversity.	***X No Yes (please describe):***
CSUSM 2: Exposure to and critical thinking about the interrelatedness of peoples in local, national, and global contexts.	***XNo Yes (please describe):***

Part D: Course requirements to be met by the instructor.

Course Requirements:	How will this requirement be met by the instructor?
Course meets the All-University Writing requirement: A minimum of 2500 words of writing shall be required for 3+ unit courses.	Students will be expected to write out solutions to problems, write programs and document the programs.
All courses offered in area B4 must have a prerequisite	CS 105 has the Completion of the Entry-Level

of at least intermediate algebra and must use a level of mathematics beyond that of intermediate algebra. No remedial algebra courses (e.g., Math 10, 20, and 30) can be used to satisfy this requirement. Even if a course has intermediate algebra as a prerequisite, it will not satisfy the Quantitative Reasoning Requirement unless it also meets each of the following three conditions:	Mathematics (ELM) requirement.. ELM is intermediate algebra. Notions above the level of intermediate algebra: conversion between non-decimal numbers systems, Boolean algebra and the algebra of other unary and binary operators introduced in class.
• It must focus on the use of mathematical language and formal reasoning in a variety of diverse disciplines, using a broad range of examples.	The mathematical language used includes: notions of numerical systems, correct application of algebra and geometry. Formal reasoning includes using the ideas that: in an assignment statement the right-hand side of the assignment operator must be evaluated first before assigning this value to the left-hand side of the assignment operator; classes and objects; iteration can be count-based or logic-based; the result of a conditional statement is true or false. These ideas are used for each step in an algorithm as a student proceeds to implement a program as the solution to the problem given. The language and reasoning are applied to all disciplines that require of quantitative, consistent solutions. This helps students to understand the broad application of the subject.
• It must provide some historical perspective on the role which this approach has played in the development of human knowledge and of our understanding of the world.	Students are introduced to the development of computers. First users of computers. The mathematical foundations of Computer Science. The first code, ASCII, used to exchange information. The first programming languages. How to multimedia to design and implement solutions to problems. Applications that have had a significant impact in how human knowledge is disseminated.
• It must demonstrate a variety of methods, such as the use of abstract symbols, of numeric techniques, of logical reasoning, of geometry, etc.	Methods demonstrated include: usage of variables to represent quantities; usage of classes and objects that generate graphics and sound; usage of flowcharts to understand the logic of a solution; usage of a set of principles which are used to justify moving from one statement to the next in a program; usage of functions to show relationships between variables; usage of test cases to verify the logical correctness of a program and correct any errors; usage of compilers to verify the syntactical correctness of a program and correct any errors.
A **statistics component** may be included which must:	
• Develop the students' ability to comprehend the power and broad utility of the fundamental mathematical models presented, rather than merely teaching rote statistical skills; and	N/A to CS105
• Must indicate applications to several areas.	N/A to CS105
A **computer science component** may be included which must:	

• Teach a computer language that is suitable for use in diverse areas;	The programming language used that is suitable for use in a wide variety of areas is Alice. Students will learn the basic constructs that are used by a high-level programming languages to generate graphics and sound. Problems assigned include using basic functions to draw lines and curves and how these concepts are applied in physical, biological and scientific processes. The generalization of flipping and rotation using basic vectors applied to objects. Flipping, rotation, and zooming can be expressed as multiplication of the components of the original point's coordinates by vectors in the resulting image. For example, a horizontal flip of a coordinate (x,y) can be expressed as a projection of the following: . The original x component multiplied by the vector (-1, 0). We will refer to this basis vector as xBasis. . The original y component multiplied by the vector (0,1). We will refer to this vector as yBasis. Similarly, zooming a coordinate (x,y) by a zoom factor z can be represented int terms of the xBasis = (z,0) and the yBasis = (0,z). Projection matrices can also perform rotations. Questions to answer: What is the effect if you have a 45 degree counter-clockwise rotation? Is the resulting image the same size as the original? What can we conclude? Other problem assigned is related to parabolas and gravity. Model bouncing balls by adding a rate of decay. Another problem is to design a 1-Bit Full Adder constructing logic gates using Boolean Algebra and Truth Tables.
• Teach this language in such a way that the student is led to a fundamental understanding of the nature of problem solving by combining data structures with algorithms; and	The approach to teaching a programming language is to start with the most simple concepts and constructs in a programming language to keep building on these concepts and constructs by combining them in more complex structures. The final product is a program that is the code for the steps in an algorithm to solve a problem given a set of specifications.
• Provide fundamental skills in the use of computers for the application of university level quantitative methods to the solution of problems in many diverse areas.	The course is taught in a computer laboratory where students are required to apply the concepts and methods learned to write programs to solve a given problem. given problem. The solution is implemented by designing, coding, compiling, debugging, documenting and testing a program written in high-level language such as Alice. Problems assigned are in many diverse areas. Learn how to apply basic functions to draw objects using lines and curves and physical, biological and scientific processes to which they can be applied. P The generalization of flipping and rotation using basic vectors applied to objects. Flipping, rotation, and zooming can be expressed as multiplication of the components of the original point's coordinates by vectors in the resulting image. For example, a horizontal flip of a coordinate (x,y) can be expressed as a projection of the following: . The original x component multiplied by the vector

	(-1, 0). We will refer to this basis vector as xBasis. The original y component multiplied by the vector (0,1). We will refer to this vector as yBasis. Similarly, zooming a coordinate (x,y) by a zoom factor z can be represented int terms of the xBasis = (z,0) and the yBasis = (0,z). Projection matrices can also perform rotations. Questions to answer: What is the effect if you have a 45 degree counter-clockwise rotation? Is the resulting image the same size as the original? What can we conclude? Other problem assigned is related to parabolas and gravity. Model bouncing balls by adding a rate of decay. Another problem is to design a 1-Bit Full Adder constructing logic gates using Boolean Algebra and Truth Tables.

CS 105 Media-Propelled Computational Thinking

Instructor: Rikki Fletcher

IMPORTANT REMINDER

Your enrollment in this course is contingent upon attendance during the **first two weeks**. Absence without prior arrangement with the instructor *may* be considered a voluntary disenrollment.

DISABLED STUDENT SERVICES

Students with disabilities who require reasonable accommodations must be approved for services by providing appropriate and recent documentation to the Office of Disabled Student Services (DSS). This office is located in Craven Hall 5205, and can be contacted by phone at (760) 750-4905, or TTY (760) 750-4909. Students authorized by DSS to receive reasonable accommodations should meet with me during my office hours in order to ensure confidentiality.

OFFICE HOURS

The following are my office hours for Spring 2012. In addition, please see me or email via Moodle to schedule an alternate appointment.

Office Location	Office Hours	Phone Number
Sci Hall II Rm 227 Or Sci Hall II Rm 302	14:30 – 15:45 & 17:30 – 18:00 Tues and Thurs Other hours by appointment	N/A use email

COURSE DESCRIPTION

This course is an introduction to the basics of programming through multi-media. Programming languages such as Java, Jython or Python are studied and programming techniques are used emphasizes programming methodology and problem solving to first use basic functions that draw objects using lines and curves and later to explore familiar physical/biological/scientific processes. Mathematical competence necessary for academic success will be enhanced.

COURSE OBJECTIVES

To understand fundamentals of mathematics and programming

To become familiar with concepts, symbols, computations in mathematics and how they relate to programming using multimedia approach.

To enhance students' critical and analytical thinking skills

STUDENT LEARNING OUTCOMES

Explain and apply a variety of fundamental mathematical concepts, symbols, computations and principles via Alice an interactive programming environment

Apply critical and analytic thinking to presented problems

Use flowcharts for solving a given problem and correctly implement those solutions via an animation

Display effective communication both written and spoken.

Description of solutions to a problem given in assignments, quizzes, exams and laboratory work

Use pseudocode or a flowchart to design a solution

Translate their solution to code using a programming language

Document the program for other users to enhance and maintain

Determine what is being asked of them, think about what method/procedure of solution is appropriate, and properly implement that method/procedure

Understand Concepts:

> Numeric bases. Converting from one numeric base to another numeric base.
>
> Boolean algebra used in programming constructs.
>
> Analytic and geometry knowledge to think and reason in terms of 3-D.
>
> Classes and objects to implement graphics and sound.
>
> Basic constructs to write a program that is syntactically and semantically correct: decision statements, iteration statements, functions.

Understand Symbols:

> Flowcharts, scripts, story boards, arithmetic operators: binary +, unary -, binary -, binary *, binary /, binary %, logical operators: &&, ||, !; relational operators: >, >=, <, <=, ==, !=; assignment operators : =, +=, -=, *=, /=, %=.

Understand Computations:

> Programs are written, compiled, debugged and tested for correct application of constructs.

Understand Principles:

> Planning a solution for a given problem, designing the solution using a flowchart, script or story board to review the logic of the solution, translating the solution into code following the syntax of a programming language, compiling the code to remove any syntactic errors, testing the program with several test cases to correct logic errors, documenting code for ease of maintenance.

ALL UNIVERSITY WRITING REQUIREMENT

The writing requirements for this class will be met as described in the homework and project assignments. Every course at the university, including this one must have a writing requirement of at least 2500 words.

PRE-REQUISITE FOR CS 105

Completion of the Entry-Level-Mathematics (ELM) requirement.

TEXTBOOKS:

A. *Alice in Action*, by Joel Adams; Published by Cengage Learning
B. *Using and Understanding Mathematics,* by J. Bennett and W. Briggs, Published by Addison Wesley

MOODLE:

Our campus provides support for e-learning through Moodle. Moodle can be accessed via **http://cc.csusm.edu**. You will need a campus account to log into Moodle. To activate your campus account, bring a picture ID to ACD 202 or Kellogg Library 2000.

ASSESSMENT

You will be expected to write out solutions to problems showing the logic and flow of information of your solution, coding the solution using a programming language, and documenting the program in detail so that other users, in particular other students, read and understand the program for further development.
Your grade for the class will be assessed based on homework assignments, in-class exercises, exams and projects. The exam will be closed-book. No access to cell phones, calculators, computers, or individuals other than the proctor is allowed during exams; **NO MAKE-UP EXAM** will be given except under the most extraordinary of situations. Your in-class exam is scheduled on May 3rd. The final exam is scheduled on Tuesday, May 15th. (11:30 - 1:30 PM) Mark your calendar now. The following table indicates the weight for each portion of your work towards your final grade.

Category:	Homework	Exam	First Project	Final Project	Classwork & Quizzes
Weight:	20%	20%	15%	25%	20%

Your letter grade for the class will be based on the following scale:

Overall %	>= 92	>= 89 < 92	>= 86 < 89	>= 82 < 86	>= 79 < 82	>= 76 < 79	>= 70 < 76	>= 60 < 70	<60
Grade:	A	A-	B+	B	B-	C+	C	D	F

ACADEMIC HONESTY

Students will be expected to adhere to standards of academic honesty and integrity as outlined in the Student Academic Honesty Policy. All written work and oral presentation assignments must be original work. All ideas/material that are borrowed from other sources must have appropriate references to the original sources. Any quoted material should give credit to the source and be punctuated with quotation marks.

Students are responsible for honest completion of their work including examinations. There will be no tolerance for infractions. If you believe there has been an infraction by someone in the class, please bring it to the instructor's attention. The instructor reserves the right to discipline any student for academic dishonesty, in accordance with the general rules and regulations of the university. Disciplinary action may include the lowering of grades and/or the assignment of a failing grade for an exam, assignment, or the class as a whole. Incidents of Academic Dishonesty will be reported to the Dean of Students. Sanctions at the University level may include suspension or expulsion from the University.

ADDITIONAL POLICIES

1. Computer access during lecture is restricted to class related activities.
2. Cell Phones: Please silence cell phones, including text. If an emergency situation exists, please respond to any phone message outside of the classroom.
3. Email: Use the Class Mail option on **Moodle** for all class related emails.
4. Homework: All homework assignments will be posted and collected via Moodle. Late submissions are NOT ACCEPTED. While discussions among students are encouraged all assignments are to be completed individually. Questions concerning homework assignments shall be directed to the instructor or posted on Moodle discussion forums for feedback from other students. Grades for assignments will be released on Moodle.
5. Grade Appeal: Any discrepancy on grades shall be submitted to the instructor **within one week** from the day that the grade is released (not the day you check it)
6. Flash Drive: It is recommended that you use a flash drive or other external storage system to save all your coursework.

TENTATIVE SCHEDULE, *SUBJECT TO CHANGE*:

Week	Dates	Subjects	Book Chapters
1	01/24	Introduction	N/A
1	01/26	Overview / Getting started with Alice	A1
2	1/31 & 02/02	Program Design and Flow Charts	A1
3	02/07 & 02/09	Program Design; Setting up an initial scene, Thinking in 3D	B11
4	02/14 & 02/16	Classes and Methods, The camera	A2
5	02/21 & 02/23	Methods and parameters	A2
6	02/28 & 03/01	Story Boards	A3
7	03/06 & 03/08	Flow control	A4

8	03/13 & 03/15	Project Presentations (Attendance mandatory)	
9	03/22 & 03/24	**Spring Break, campus closed**	
10	03/27& 03/29	Fundamentals of binary math	B1
11	04/03 & 04/05	Event handling	A6
12	04/10 & 04/12	Boolean Evaluation	A4
13	04/17 & 04/19	Loops	A4
14	04/24 & 04/26	More Functions and Control Statements	A Appendix A
15	05/01	Review	
15	05/03	**Exam**	**All covered material**
16	05/08 & 05/10	Final Project Presentations (Attendance mandatory)	
	05/15	**Final Exam** 11:30-1:30	