

**Course Syllabus – Math 107**  
**Introduction to Scientific Programming 107 (3 Units) &**  
**Introduction to Scientific Programming Lab 107L (1 Unit)**  
**Fall Semester 2005**

**Section 61415/MW 6:30-7:50 P.M. & Section 55859/MW 8:00 - 9:20 PM**  
**Room I207M**

**Instructor:** Steve Horton  
**Phone:** (858) 922-9989  
**Email:** shorton@sdccd.edu  
**Mailbox:** H207

**Text and Materials:** Java: An Introduction to Computer Science & Programming,  
3rd Edition (2004), by Walter Savitch

**Course Description:**

This course is an introduction to mathematical and scientific problem-solving on a computer; focusing on designing algorithms of a high level programming language. Extensive programming is required. Students are expected to plan and write programming projects with documentation. This course is recommended for students transferring to majors in Computer Science and /or mathematics. (FT). Transfer Credit: CSU and/or private college/university. UC Transfer Course List. Mathematics (MATH) 107 combined with Computer and Information Sciences (CISC) 185: maximum credit, one course

**Course Objectives:**

Upon successful completion of the course the student will be able to

1. Develop, debug, and execute programs solving problems from a wide array of mathematical, scientific and economic applications.
2. Construct programs using user-defined procedures and functions, and incorporate parameter portability.
3. Implement decision structures and nested decision structures.
4. Organize programs so that the output is in a user-friendly fashion giving a menu for options.
5. Implement iterative algorithms using looping structures, including nested loops.
6. Determine the output of programs and procedures by hand tracing segments of source code.
7. Construct programs using one-, two-, and three-dimensional arrays.
8. Implement basic sorting and searching algorithms, including insertion sort, selection sort, bubble sort, and linear search.
9. Construct programs using text file processing.
10. Construct programs for applying and processing records.

**Prerequisite:**

MATH 096, with a grade of "C" or better, or equivalent, or Assessment Skill Level M50.

**Corequisite:**

MATH 107L – this is a 1-unit lab course to be taken concurrently with Mathematics 107. Extensive programming is required. You will receive a **separate grade** in Math 107 and Math 107L. Students are expected to plan and write programming projects with documentation. This course is recommended for students transferring to majors in Computer Science and/or mathematics. (FT). Transfer Credit: CSU and/or private college/university.

**Advisory:**

ENGL 056 with a grade of C or better, or equivalent.

**Academic Accommodation:** Any student who may need an academic accommodation should discuss the situation with me within the first two weeks.

**Important Dates:**

August 31st	First day of class
September 5 <sup>th</sup>	Labor Day (No class)
November 4th	Withdraw deadline
December 16th	Final exam (Last day of class)

**Class Policies:**

**Attendance:** Attendance is critical to teaching and learning. You will fall behind in acquiring course content and skills if you do not attend class regularly. Therefore, there is a maximum of three un-excused absences before being automatically dropped from the class.

**Conduct:** Students are expected to respect and obey standards of student conduct while in class and on the campus. In addition, please no food in the lab, turn off any cell phones, and no computer games during class time. Finally, cheating or any form of academic dishonesty is not acceptable and will be reported to the Student Affairs Officer for disciplinary action (Policy 3100).

**Late Assignments & Make-up Exams &:** A positive outcome is my goal for every student of this class. Experience has shown that falling behind makes it very difficult to achieve this goal. As a result, there is no late work accepted except for legitimate medical or emergency reason and is completely by instructor discretion. For any exam (excluding the final) with an approved legitimate reason, you must notify me at least one week in advance and arrange to make up the exam on my limited schedule. There is no make-up for the final exam.

**Incompletes:** To receive an incomplete, you must be passing this course and unable to take the final examination.

**Course Evaluation:**

The evaluation in this course is based on two quizzes, a final, eight “lab” programming projects and a final programming project. Seventy percent of the grade in this course is derived from programming based projects. To become a proficient programmer, you must devote a great deal of effort to designing, coding, and debugging programs. It is normal to spend many non-class hours every week programming for this course, so do not be dismayed if you find yourself doing this. You must do your own work, although you may ask questions of other students or the instructor. You must understand and be able to explain your own code to the minutest detail. If you can not, then it will be assumed that someone else authored your code, and you will risk not receiving credit for the project.

**Course Grade:**

Final course grades are computed based on the following weighting:

**Math 107**

<b>Quiz #1</b>	<b>30%</b>
<b>Quiz #2</b>	<b>30%</b>
<b>Final Exam</b>	<b>40%</b>

**Math 107L**

<b>Labs (8)</b>	<b>80%</b>
<b>Final Project</b>	<b>20%</b>

See the attached “Tentative Course Outline” for exam and lab due dates.

**Grade Scale:**

**A:** 90% or above, **B:** 80-89%, **C:** 70-79%, **D:** 60-69%, **F:** 59% or lower.

**Extra Credit:**

Extra credit will be given for any student who turns in an extensive “My Coding Conventions” reference document at the end of the term.

**Extra Help:**

I will be available after class to answer questions. You are strongly encouraged to see me during this time. If you have a conflict at this time, please see me for other arrangements. Also, help can be obtained at the tutor center in K211.