

MATH 212 Computational Mathematics II

1. Catalog Description

MATH 211, 212 Computational Mathematics I, II (4) (4)

Fundamentals of procedural programming in C/C++ and selected applications to problems in integral and differential calculus, matrix analysis, geometry, and physics. 4 lectures. **MATH 211** prerequisite: MATH 141 or consent of instructor. **MATH 212** prerequisite: MATH 211.

2. Required Background or Experience

Math 211.

3. Learning Objectives

The student should:

- a. Understand reference variables and passing parameters by reference.
- b. Understand pointers and dynamic memory allocation.
- c. Understand the basic concept of a class, and its use with pointers in implementing data structures such as linked lists and stacks.
- d. Understand the C++ preprocessor and the use of multiple files and header files.
- e. Use programs to solve a variety of mathematical problems that require the use of more complex data structures.

4. Text and References

A text containing detailed descriptions of programming projects will be prepared by the Mathematics Department. Students will also use a C++ reference.

5. Minimum Student Materials

Paper, pencils, notebook and access to university computing resources.

6. Minimum University Facilities

Classroom with ample chalkboard space for class use.

7. Content and Method

	<u>Lectures</u>
1. Reference variables and pass by reference	4
2. Pointers and dynamic memory allocation	8
3. Classes and data structures	16
4. The preprocessor and multiple files	<u>6</u>
Total	34

8. Methods of Assessment

The primary method of assessment is assigned programming projects. In addition, written exams on programming and problem-solving will be used. The emphasis should be on the whole problem-solving process, not just on writing correct programs. All instructors will use the same assessment methods, though individual instructors may weigh the assessment methods differently. The learning objectives are listed below. The numbers in parentheses following each objective indicate the appropriate assessment method according to the scheme 1 = programming projects, and 2 = written exams.

The student should:

- a) Understand reference variables and passing parameters by reference. (1,2)
- b) Understand pointers and dynamic memory allocation. (1,2)
- c) Understand the basic concept of a class, and its use with pointers in implementing data structures such as linked lists and stacks. (1,2)
- d) Understand the C++ preprocessor and the use of multiple files and header files. (1,2)
- e) Use programs to solve a variety of mathematical problems that require the use of more complex data structures. (1)