Prof. Burt Rosenberg
Ungar 523, x–2575
Office hours: TBA.

MTH 220–T/MTH 317–T (Fall, 1994)
Time: Tu., Th. 5:00–6:15 P.M.
Room: MM 100

- **Goals:**
  The goals of this course are,

  1. Programming Practice: data structures, ADT, pointers, modules, objects and debugging.
  2. Program Correctness: loop invariants, assertions, formal logic, and test strategies.
  3. Program Efficiency: classic algorithms for sorting and searching, recursion and the dogma of the "Big Oh".

- **Prerequisites:**
  Students who have completed Math 110 and have taken Calculus I are prepared to take this course. We shall review the necessary elementary knowledge of Pascal, algebra, set theory and formal logic.

- **Text:**

  1. CONDENSED PASCAL by Doug Cooper.
  2. PROGRAM DERIVATION by Geoff Dromey.

- **Computer Facilities:**
  You will be working on the Information Resource's VAX computer in the VMS operating system. An account will be set on this computer for each student enrolled in this course. Handouts about log in, commands, editing and email are provided by Information Resources.

  The VAX/VMS is superior to a home computer in two ways:

  1. The class is actually being held in cyberspace.
  2. Big computers are cool.

- **Programming Assignments:**
  There will be about eight programming assignments. They will take between a week and two weeks each, and they must be turned in on time. Programming assignments and tests will account about equally for your grade. You cannot pass

this course if you don't learn how to program. Hence, you cannot pass this course if you don't do your programming assignments.

- **Grading Programs:**
  Programs are difficult to grade. Your documentation will aid the grader in assigning points, especially when the program is not completely working. Programs which do not compile are treated harshly. Many students are under the delusion that only the compiler stands in the way of successful completion of their programming assignments. In fact, semantic correctness is miles down the road from syntactic correctness, hence a non-compiling program is certainly non-functional as well.

  Programs will be graded as follows,

  - ⋆ 5 — Program is excellent.
  - ⋆ 4 — Program is satisfactory.
  - ∘ 3 — Program runs but is not satisfactory. It may be that it breaks too easily, be incomplete or be badly written.
  - ∘ 2 — Program compiles but does not work.
  - ◇ 1 — Program does not compile.
  - ◇ 0 — No credible effort.

- **The Honor Code:**
  The University of Miami has an Honor Code, and I intend to support it fully in this class. It is not always clear what is acceptable and what is not when student's share ideas and work. You can discuss approaches to a problem and general observations about computers and programming. When you code, however, you code alone. Document your compliance with the Honor Code on each assignment by copying it into the documentation of your programs.

- **Tests:**
  There will be a midterm in class and a final during the usual final period. Programming assignments and tests will count about equally towards your grade.