Some Notes on AM 40 (Introduction to Non-Numerical Computer Science) and AM 101-102 (Foundations of Non-Numerical Computer Science)

1. Similarities

Both courses are laboratory courses stressing programming proficiency beyond the ability to use the computer merely as a "number-crunching" tool for scientific problems. They are, however, not "programming" courses, since an understanding of the underlying functional structure of the computer and non-numeric applications are emphasized in order to place the role of computers and computer science in perspective. Both courses require an inordinate amount of time: in excess of 10 hours/week of class and laboratory time (i.e., on the average, 2 hours per day, every day, for a semester!). You will prepare programs (homework) for our IBM System/360 Model 50 (a computer with speed on the order of 200,000 operations/second and costing approximately $1,500,000). These homework problems constitute 50% of the grade in both courses, and must be documented in considerable detail, as is customary for laboratory reports.

2. Differences

a) AM 101-102 is a full-year course for prospective computer science "majors" or dedicated computer users. Most of the first semester is spent in attaining a high level of programming competence (in "machine" or "assembly" language) and in the logical design of a computer. The second semester is devoted primarily to a study of the theory of programming languages (FORTRAN, ALGOL, SNOBOL) and their translators.

Prerequisites: preferably at least junior standing and some training in mathematics, logic, or axiomatic (formal) systems, though no specific knowledge is required. (Freshmen and sophomores must see me for permission.)

Time requirement: 15-20 hrs/wk, including class and supplementary lecture time.

Meeting time: probably MWF in the afternoon.

b) AM 40 is a more cursory overview of computer science for people who want to invest only one semester, and that at an easier (though hardly leisurely) pace than AM 101-102. As a terminal course with greater breadth and less depth than 101-102, it covers logical design of the computer and programming in machine (and assembly) language, FORTRAN and SNOBOL. A variety of applications will be mentioned and some attempt will be made to deal with the socio-economic implications of the computer revolution. In a sense AM 40 can be viewed as a "technical arts" course for liberal arts majors; unlike a computer appreciation course, however, it still demands the attainment of a measure of programming proficiency.

Prerequisites: high-school algebra.

Time requirement: 10-15 hrs/wk, including class and supplementary lecture time.

Meeting time: MWF, 10 am, Metcalf Auditorium.
3. Alternatives to AM 40 and AM 101-102 for:
   a) Science and engineering problems: FORTRAN programming taught as a non-
      credit service course by the Computing Laboratory.
   b) Graduate language exam: as above.
   c) Numerical analysis and computation (FORTRAN): AM 20, Professor Feldstein
   d) Computers in linguistics (PL/1): Linguistics 170, Professor Kucera
   e) Computers in psychology (PDP8-s): Psychology 118, Professor Millward

4. Ground rules for AM 40 and AM 101-102

   a) Grades: 50% on homework problems (4 major problems in AM 101, 5 or 6
      minor ones in AM 40): This weighting implies the necessity of working independ-
      ently; i.e., no collaboration. A homework grade is determined 50% by the pro-
      gram itself, which is checked and graded by the computer, and 50% by the docu-
      mentation, checked and graded by the graders. 10% and 30% of the final grade
      are allotted to a mid-term and a final exam. respectively; 10% is miscellaneous.

   b) There is no text.

   c) Consulting hours: Graders will be available, according to a schedule
      which you will receive shortly, for some 30 to 40 hours a week to help you with
      your problems (though not to do your thinking or your programming for you) and
      to teach supplementary material as required (usually in the evening). You are
      expected to avail yourselves of their services and to submit work to the
      computer on a near-daily (nightly) basis (programs cannot be "debugged" on a
      cram basis).

   d) Auditors are subject to the same rules, perils and privileges as are
      regular members, and must maintain a B average.

   e) Although the size of the classes tends to make it difficult, I invite
      and expect class participation and discussion. Similarly, my door is always
      open for constructive feedback and suggestions.