SOME THOUGHTS ON THE TEACHING OF MATHEMATICS

IGOR RIVIN

1. What is the problem?

A mathematics professor in a public university has many responsibilities. These include research, administration, and teaching. Teaching, in turn, include "specialized" teaching (to mathematics majors and graduate students) and "service" teaching: teaching mathematics to first and second year students. These thoughts will center primarily on service teaching, which, for me, combines some of the most exciting and some of the most depressing aspects of my job.

1.1. **The product.** Why depressing? Consider: the vast majority of service courses are concerned with differential and integral calculus and linear algebra. These are both rather deep subjects, as evidenced by the fact that mathematics had been practiced for thousands of years by rather talented people before the basic principles of the calculus were laid down in the late seventeenth century. It took another two hundred years of extensive work to make the foundations of the subject truly solid. Linear algebra, as used to day, is an even later bloomer. The current machinery of matrices and linear transformations was not put into a truly modern form until the beginning of the twentieth century. It must be admitted that the subjects in questions are quite deep and require some considerable technique to use successfully.

1.2. **The consumers.** *Who* are we teaching them to? In a public university (such as Temple) our students are, in the main, somewhat above-average products of the US public secondary education system, which means that they techincal ability in manipulation is already quite severely taxed by arithmetic with fractions. Their abstract reasoning skills are essentially nonexistent, and the very concept of proof is foreign to them.

Key words and phrases. mathematics teaching, constructive methods, differential and integal calculus, linear algebra, service teaching.

The author is supported in part by the NSF DMS. These thoughts are very much in a preliminary DRAFT form.

1.3. **The results.** A consequence of all this is that it is well-nigh *impossible* to teach them what we purport to teach them: higher mathematics presupposes a certain level of abstraction, and even if we commit the crime of forgetting that, and define calculus as "a collection of computational techniques without understanding," the students' technical weakness renders even that aspect essentially worthless. They *can not* compute. The result is that our calculus and linear algebra classes consist of a collection of trivial examples which the students must memorize by rote. This has the consequences of not teaching the students anything except the fear and hatred of mathematics. There is more still: the majority of the students never use calculus in their future lives (small wonder, since they don't actually know any, as discussed above, but they never had any intention of using advanced mathematics even *before* taking the courses). They are required to take the courses because of the (not unreasonable) belief that mathematics should be a part of ever college-educated person's intellectual make-up. The result is that the *loathing* of mathematics is part of the intellectual makeup of a sizeable majority of Americans. The amazing (and exhilarating) observation is that despite all of the above, some students actually manage to understand something of the subject... The exhilaration is, however, tempered by the thought of the huge amount of wasted time and by the thaughts of what these talented students could achieve if taught properly.

2. What can we do?

What, then, is the solution? We could drop the distribution requirements of Mathematics (and I could easily see this happening), but the fact of the matter is that the ability to reason logically and abstractly really should be someone everyone takes from their higher education, and we, as mathematics educations (which we are, even if the term does produce a visceral reaction in most people) to instill it in our students. How? The first step is a step back – a step back from "higher" mathematics – the mathematics of infinite and the infinitesimal, and to go back to conceptually simpler forms of mathematics.

In the (not so distant) past, Euclidean geometry was such a subject, taught exactly for the above-stated reasons (the fundamental concepts – of line, circle, distance, *etc.* are quite intuitive, while the basic components of mathematical reasoning are all present). I would not advocate the return to this, however, because, firstly, the subject has been dead for several hundred years and secondly, it is quite far from the modern American experience.

Instead, it is my opinion that we should start at the very beginning – with reasoning (logic) and counting (which means naive set theory and combinatorics and graph theory) and probability. These subjects are ever more visibly important in our lives due to the ubiquity of computers. These subjects are both easier to learn and more immediately rewarding for the students. I addition, there is another principle which can be used to clear at least some of the mush out of the students' minds. That principle is: "a computer program *is* a proof." This correspondence (though far from perfect) allows us to make mathematics hands on (programming a computer gives very rapid feedback, both positive and negative), closely related to the students experience and visibly "useful." (of course, the real utility of mathematics lies much deeper, but . . .).

One problem with this approach is the potential need to waste a lot of time introducing students to the subtleties and idiosyncrasies of some (possibly proprietary) programming languague or a scientific computing system. It is important to start the students off on a mathematically clean system, preferably running on a mathematical *abstract machine*. Luckily, such a system is available and has been used for over twenty years in *Computer Science* education at MIT, University of Indiana, and many other schools. This is the SCHEME programming language.[1] The justly acclaimed book by H. Abelson and G. J. Sussman ("Structure and Interpretation of Computer Programs") introduces the fundamentals of computer programming, and together with a companion book on "Structure and Interpretation of Mathematics"[2] this would constitue the core of a modern introduction to Mathematics.

3. Possible Objections

One could forsee some objections to the above program:

3.1. **But what about calculus?** There is no suggestion of eliminating calculus completely from the University curriculum: the students of the sciences and engineering (not to mention mathematics) do need to be acquainted with it. However, it would enter somewhat later, when the students are more mathematically mature, and thus more

[1]See `http://www.scheme.org` for more details.

[2]Writing such a book is a project I am very keen on, but this has to be done in parallel with using these ideas in teaching – the book of Abelson and Sussman was circulated as lecture notes for several years

capable of actually understanding something of the subject. It is true that a lot fewer students will be *required* to study calculus, but, on the other hand, the number of people studying *Mathematics* may actually increase.

3.2. **But what about the teachers?** It has been my observation that many of the TAs, having been educated in a "traditional" way are a little rusty on "finite" mathematics. Same goes (in spades) for many of the professors. Many faculty members and graduate students have no familiarity with computer programming at all. The first problem is easily fixable, the second slightly less so, but my contention is that anything which is teachable to freshmen should be even easier to teach to faculty (and is no less useful to them).

Department of Mathematics, Temple University
E-mail address: `rivin@math.temple.edu`