

Abstract for a Short Lecture

Math with Programming – Shaken or Stirred?

Miroslav Lovric, McMaster University

In this presentation I will report on the classroom experience (from both students' and instructor's perspectives) in an advanced mathematics problem-solving course (called Math 3G3), in which students were asked to investigate mathematical problems using computers. In this case, 'using computers' means writing one's own computer code, as opposed to using ready-made software where users manipulate parameters, change settings, etc., but are not (in most cases) aware of underlying mechanisms that control the software.

Students were given freedom in choosing a programming language (most common choices were Maple, MATLAB, Python and C++). However, in order for all of us to communicate, I asked students to use a pseudocode, for instance on assignments and tests. Even though students in Math 3G3 were good mathematics students (all majors in mathematics), programming turned out to be quite challenging. I will comment on the conflicts between my expectations and my students' experiences, and on a number of cognitive situations that emerged, all of which will help me adjust the course the next time I will teach it.

Since Math 3G3 included paper and pencil problem-solving activities as well, I was able to contrast students' thinking, approaches to solving problems and attitudes in using two distinct media (paper and computer), and came up with interesting and useful insights. I will speak about the tension between the paper and the computer as problem-solving tools, and offer suggestions for a possibly more smooth transition between the two media.

Although the context of Math 3G3 is upper-year university mathematics, I believe that the observations and messages are relevant for almost any programming-intensive mathematics course in a tertiary institution.

Miroslav Lovric
Department of Mathematics and Statistics
McMaster University, Hamilton, Ontario, Canada
lovric@mcmaster.ca