Selected Passages from Correspondence with Friends 10 no. 1 (2022), 1–4. ISSN 2054–7145

IMPLEMENTATION OF THE KID KRYPTO CONCEPT

ALEXANDRE V. BOROVIK

My colleagues asked me to make available on the Internet my paper

> Alexandre V. Borovik,
> Implementation of the Kid Krypto Concept:
> Computer Assisted Teaching of Number Theory and Cryptography.
> MSOR Connections 2 no. 3 (2002), 23–25.

MSOR stands for Mathematics, Statistics, and Operation Research, this was the Newsletter of the LTSN Maths, Stats and OR Network. According to Wikipedia, LTSN is "Learning and Teaching Support Network"; Maths, Stats and OR Network was one of LTSN's subjects centres and was closed in 2011. † Publication of the newsletter stopped and was resumed in 2015 as an open access journal‡ However, although it continues the numbering of volume of the Newsletter, its archive contains no issues from 2011 or earlier years.

Therefore I feel that it will be appropriate to place my paper, as it was published in 2002, on the next three papers here.

Alexandre Borovik
29 August 2022

Disclaimer

The author writes in his personal capacity and the views expressed do not necessarily represent position of his (former) employer or any other person, corporation, organisation or institution.

Email alexandre >>>at<<< borovik.net

2020 *Mathematics Subject Classification* 97D40.

†Learning and Teaching Support Network;https://en.wikipedia.org/wiki/Learning_and_Teaching_Support_Network.

‡https://journals.gre.ac.uk/index.php/msor/about/editorialPolicies#openAccessPolicy.

Implementation of the Kid Krypto Concept

Computer Assisted Teaching of Number Theory and Cryptography

Alexandre V Borovik
UMIST

a.v.borovik@umist.ac.uk

I wish to start by briefly outlining a conceptual framework for my views on the didactical efficiency of computer assisted methods in teaching mathematics. According to David Mumford [9, p199], mathematics is "the study of mental objects with reproducible properties".

Learning mathematics has at least two intertwined aspects:

- Interiorisation of other people's mental objects.
- The development of the reproduction technique for your own mental objects.

Assessment of software for teaching and learning mathematics should address these two issues, interiorisation and reproduction.

There is a natural hierarchy of reproduction methods. A partial list includes: proof; algorithm; symbolic and graphic expression. I wish to clarify that reproduction is more than communication: you have to be able to reproduce your own mental work for yourself.

Interiorisation is less frequently discussed; for our purposes, we mention only that it includes visualisation of abstract concepts and transformation of formal conventions into psychologically acceptable "rules of the game". At a more mundane level, you cannot learn an advanced technique of symbolic manipulation without first polishing your skills in more routine computations to the level of almost automatic perfection. Interiorisation is more than understanding; to handle mathematical objects, one has to imprint at least some of their functions at the subconscious level of one's mind.

It is time to give a few examples. For some years I taught courses in mathematical logic based on two well-known software packages: SYMLOG [10] and TARSKI'S WORLD [1] (reviews: [3], [6]). SYMLOG used a very poor DOS command line interface, while TARSKI'S WORLD (TW) very successfully exploited the graphic user interface of Apple and Windows for the visualisation of one of the key concepts of logic, a model for a set of formulae, see [2] for the discussion of the underlying philosophy. Also, TW made a very clever use of games for explaining another key concept, the validity of a formula in an interpretation (although the range of interpretations was limited [6]). However, when it came to a written test, students taught with SYMLOG made virtually no errors in composing logical formulae, while those taught with TW very obviously struggled with this basic task. The reason was easy to find: SYMLOG's very unforgiving interface required retyping the whole formula if its syntax had not been recognised, while TW's user friendly formula editor automatically inserted matching brackets. Although TW's students had no difficulty with rather tricky logic problems when they used a computer, their inability to handle formulae without a computer was alarming. Indeed, in mathematics, the ability to reproduce your mental work has to be media-independent. Relieving the students of a repetitive and seemingly mindless task led them to lose a chance to develop an essential skill.

A brief summary: the aim of computer assisted mathematics teaching is not to make students' life easier. It should be used only when it introduces a new quality to the teaching and allows students to achieve something which cannot be done by traditional methods.

Kid Krypto

The first year introductory course in Number Theory and Cryptography, as it is currently taught at UMIST, was developed by the author over a number of years and brought in its current form in collaboration with Richard Booth. The present paper borrows a lot from [4] where the interested reader can find more details about that particular course. The computer assisted teaching of number theory is not a new idea; I will mention here only one excellent book on the subject [5]. By the very nature of elementary number theory, software packages like MATLAB and MATHEMATICA are used as glorified calculators. Therefore the general methodology of teaching is essentially platform independent. At this background, the new element of the course is that it is built on the concept of "Kid Krypto" as formulated by Neal Koblitz, one of the creators of modern cryptography. Koblitz defines Kid Krypto [8, p17] as

> *"Development of cryptographic ideas that are accessible and appealing (and moderately secure) to those who do not have university-level mathematical training".*

In our course, we use this principle as the way of justifying the need for mathematical rigour. In effect, we expand Koblitz's thesis to include:

> *"Proving to the students that a sound mathematical theory is required, by giving them their own experience of breaking down cryptographic systems intentionally built on a weaker mathematical foundation".*

To enjoy the full benefits of the Kid Krypto approach, we set up an unusual mode of interaction between the students: besides collaborative work in small groups, students are encouraged to attack and break each other's cryptosystems. Indeed, the students take the fate of their creations so close to heart that their true identities have to be protected by aliases, to save the less fortunate from the embarrassment of failure under their peers' sustained attacks.

Content of the course

"Number Theory and Cryptography" is a first year mathematics course; it is of introductory nature and has no dependent courses. It is small; 10 lectures (of the traditional blackboard style) and 10 computer lab sessions. The course introduces number theory through its applications in public key cryptography. It assumes nothing more than standard A Level mathematics, and every result presented in the course is rigorously proven from first principles. This is motivated by the need to be sure of our cryptosystems; if (as is explained to the students), these systems sometimes take on literal life-or-death importance, it is good to be sure of one's ground. The course covers the traditional elementary number theory, from division with remainder to Euler's and Chinese Remainder Theorems. Students also learn some material which has become a standard feature of modern introductory courses in number theory:

- The Rivest-Shamir-Adleman (RSA) public key encryption protocol.
- Miller-Rabin primality test.
- Pseudoprimes, Carmichael numbers.

The cryptographic side of the course gives the students some rather cursory familiarity with some of the fundamental concepts of cryptography: hash functions, RSA, Diffie-Hellman key exchange, and some of RSA-based protocols, such as third party certification of public keys.

The computational platform of the course is MATLAB (with its Extended Symbolic Toolbox). However, with the same ease the course can be based on MATHEMATICA, or very easily transferred to JAVA (see [7] for a project under development). Essentially, we use MATLAB as a calculator, although the polymorphism between the standard short integers, "string integers" and symbolic integers is very handy. No use is made of any advanced presentation tools. This is an intentional choice: we want our students to make on their own every small step all along the way from the mathematical concept to its working implementation. The command line interface of MATLAB forces the user to explicitly formulate everything he/she wants the computer to do. It is appropriate to mention that, besides visualisation, there is another mode of interiorisation, namely verbalisation. Indeed, we much better understand those things which we can describe in words. In naive terms, typing a command is like saying a sentence, while clicking a mouse is equivalent to pointing a finger in conversation. I want my students to speak.

How the course works

Since use of cryptography in electronic communication is the primary source of motivation, it is natural that computers are used as the communication medium of the course. Assignments are submitted via email, and some assignments require email collaboration with other students. One should not underestimate the educational value of a teacher's comments on students' errors. Usually, comments on typical errors are circulated to all the class.

The work on which the students are assessed is almost entirely practical in nature. They start with simple

experiments and eventually develop cut-down versions of the cryptosystems. At various points during the course, students are encouraged to break mathematically crippled cryptosystems introduced by the lecturer. An attack on the Kid Diffie-Hellman key exchange, based on modular multiplication instead of exponentiation, is a very instructive exercise. Students post their own keys for the cut-down cryptosystems developed, and attempt to break any poorly chosen keys of other students. Keys, marks, etc are all posted under aliases, in order to protect students' privacy. It also allows the lecturer to introduce artificially weak bogus students, as targets for the others. For example, students have the tendency not to randomise their systems and sometimes produce RSA bases like

```
100000000000000003000000700000000000000021
```

This number is a product of two primes; to break the RSA means to factorise it. For numbers as above, this can be done by mental arithmetic, thus providing another nice exercise. If no one in the class makes this mistake, bogus students come to help.

Student-centred learning is an expected benefit of computer assisted learning, and it is fully achieved in the course. Kid Krypto introduces firm and objective criteria of assessment: a key exchange protocol either works or it does not. A cryptosystem is either broken or it withstands attacks. This gives the students very efficient feedback and allows them to take some degree of control of their learning. Cryptographic protocols (certification of keys, for example) are very natural projects for collaborative work by small teams of students. The teams work at their own pace and can move ahead of the rest of the class. We have seen, on many occasions, that stronger students were actively looking for – and finding – course related material elsewhere on the Web.

Is the use of Kid Krypto justified?

Yes, we believe so. When we look at the course in the interiorisation / reproduction framework, the role of the computer assisted elements is almost entirely at the level of emotions, not cognition; they make mathematics heart-felt, justify the mathematical rigour, make the students experience mathematical problems as personal issues and set firm and demanding criteria for students' own assessment of their work. Essentially, they create a normal psychological environment for learning mathematics. We cannot expect more from a laboratory work component of a general introductory course. Its learning outcomes are intangible assets like students' enthusiasm towards mathematics or their acceptance of mathematical rigour. The core of the course is still lecture based and saturated by proofs, and the fact that it is popular with students is most encouraging.

Where else can Kid Krypto be used?

We believe that the Kid Krypto approach can be widely used throughout mathematics, computer science and communication engineering departments. For non-mathematicians, the theoretical content of the course can be diluted, allowing them to play more games with different cryptographic protocols. For example, the public key infrastructure for electronic commerce can be modeled using modular multiplication in place of exponentiation. This requires just a few simple JAVA applets, and can be easily made into an entirely on-line course.

References

[1] J Barwise and J Etchemendy, *The Language of First-Order Logic*: including the IBM-compatible Windows version of Tarski's World 4.0, Stanford, 1992

[2] J Barwise and J Etchemendy, *Computers, visualization, and the nature of reasoning*, in The Digital Phoenix: How Computers are Changing Philosophy (T W Bynum and J H Moor, eds) London: Blackwell, 1998, pp 93-116

[3] G Boolos, *Review of Jon Barwise and John Etchemendy, Turing's World and Tarski's World*, J. Symbolic Logic 55 (1990) 370-371

[4] R Booth and A Borovik, *Mathematics for information technology: the challenge of rigour*, in Snapshots of Innovation 2002, Manchester: Curriculum Innovation, 2002, pp 2-4 (ISBN 1-903640-06-7)

[5] P Giblin, *Primes and Programming*, Cambridge University Press, 1993

[6] W Hodges, *Review of J Barwise and J Etchemendy, Tarski's World and Turing's World*, Computerised Logic Teaching Bulletin 2 (1) (1989) 36-50

[7] L Hodson et al, *Number Theory I. An Independent Learning Enrichment Course*, http://math.usask.ca/encryption

[8] N Koblitz, *Algebraic Aspects of Cryptography*, Springer, 1998

[9] D Mumford, *The dawning of the age of stochasticity*, in Mathematics: Frontiers and Perspectives (V I Arnold et al, eds) Amer Math Soc 2000, pp 197-218

[10] F D Porturaro and R E Tully, *Logic with Symlog: Learning Symbolic Logic by Computer*, Prentice Hall, 1994