The interaction of algorithms and proofs in the discrete mathematics course for future engineers

Sergei Pozdniakov and Elena Tolkacheva

Abstract. The paper discusses the possibility of basing the introduction of new concepts, the deducing of their properties and the proof of theorems, based on an analysis of the algorithms associated with these concepts and theories. In this case, activity with an object comes first, which is one of the essential component of technical thinking (it can be considered as conceptual-shapely-active thinking in accordance with the work of T. V. Kudryavtsev [1]), which is not sufficiently taken into account in teaching mathematics in technical universities. It is shown that Papert's thesis to base teaching mathematics on a student's personal thinking can be applied to computer use not only at school, but also at a technical university. An example is given of two topics ("Diophantine equations" and "Continuous fractions"), which can be studied as a single section, considering different interpretations of the extended Euclidean algorithm. Based on the theoretical analysis of the given example, it is shown that when setting out the course of mathematics in technical universities it is advisable to focus on the algorithmic representation of the material. This will naturally connect the material with the activities of the programmer and thereby increase the applied character of teaching mathematics. The work was supported by the RFBR grant No. 19-29-14141

Introduction

One of the urgent problems of teaching mathematics in technical universities is the harmonization of methods of teaching mathematics with the goals of training engineers and taking into account changes in the information environment both in the student's educational environment and in the structure of the engineer's professional activity. The most entrenched tradition of building a mathematics course in a technical university is to copy the style of teaching mathematics to future mathematicians. An indicative is how most of lecturers of technical universities see the role of examples in reading a course of mathematics. Such a lecturer will first give an abstract definition of a concept, then he will prove its formal properties, then he will prove theorems and ONLY AT THE END will give an example linking a new concept with existing ideas, well-known concepts and applications. Thus, instead of giving a tool for work ("like an ax for a carpenter" according to academician Krylov [2]), the teacher builds a magnificent building of mathematics, demonstrating all its small details and admiring the logical beauty of the structure. At the same time, teachers of mathematics who work with future engineers unanimously note that the presentation of material through algorithms for actions with subject objects meets an incomparably greater response from the audience. Critics of this approach to the course of mathematics will first criticize it for the lack of a strictly logical structure and neglect of evidence. Here are a few arguments that justify this approach and show the inconsistency of such comments.

1. Substantiation of concepts

We consider one of the important arguments presented in the articles [3] by Semour Papert about changing the object basis of ideas that are formed in people's brains under the influence of the information environment. He denies the uniqueness of basing the modern mathematical culture of schoolchildren on such traditional objects as numbers and fractions and shows how studying the control algorithms for a turtle and other computer objects allows not only to form concepts using other basic ideas, but also to use them to prove statements. We will try to show that the analysis of simple algorithms can provide no less proofness than traditional sequence of theorems not related to algorithms.

2. About algorithm analysis as proof

This problem is especially interesting from the point of view of the potential ability to base reasoning not so much on formal premises as on algorithm Most theorems of mathematics are formulated in a constructive form, thereby they already give an algorithm, often far from the most effective, but which students can realize using a simple example for protocol or program for general cases. As a rule, constructive proofs are associated with cyclic (recursive or iterative) algorithms. In this case, introducing the concept of an invariant of a cycle, we can write an algorithm as a special form of writing evidence by the method of mathematical induction, and the proof of the correctness of the algorithm will actually be determined by its structure. As an example, we consider the use of the Euclidean algorithm for decomposing an ordinary fraction into a continuous one and then deriving the properties of convergents. This example is interesting in that the topic "Continuous fractions" is usually set out separately and requires a certain lecture time, while the algorithm for generating convergents is only distinguished by signs from the intermediate operations of the extended extended Euclidean algorithm. This allows not only more compactly presenting the topic, but, most importantly, showing how the same algorithm solves different problems, increasing the degree of connectivity of the material presented. The Euclidean algorithm can be written as $r_{n-1} = r_n \cdot q_{n+1} + r_{n+1}$ or $r_{n+1} = r_{n-1} - r_n \cdot q_{n+1}$ where $r_{-2} = a$, $r_{-1} = b$, and $d = GCD(a; b) = r_n$ for n such as $r_{n+1} = 0$. It can also be written in the form $r_{n-1}/r_n = q_{n+1} + 1/(r_n/r_{n+1})$, which indicates the possibility of representing the fraction a/b as an ordered set of quotients $[q_0; q_1, ..., q_n]$ which called continuous fraction. The extended Euclidean algorithm finds a particular solution to the equation $a \cdot x + b \cdot y = d$. It can be naturally obtained from the previous algorithm, considering recurrence as a vector formula $R_{n+1} = R_{n-1} - R_n \cdot q_{n+1}$, where $R_n = (x_n; y_n), R_{-2} = (1; 0), R_{-1} = (0; 1)$. The invariant of the cycle is the condition $a \cdot x_k + b \cdot y_k = r_k$. At the nth step, the $GCD(a; b)$ and its linear representation will be calculated: $a \cdot x_n + b \cdot y_n = r_n = d$. The next step is two numbers x' and y': $a \cdot x' + b \cdot y' = r_{n+1} = 0$, from which we get $a/b = -y'/x'$. Thus, the extended Euclidean algorithm can be considered as an algorithm for "folding" a continuous fraction $[q_0; q_1, ..., q_n]$ - converting it to a regular fraction $-y'/x'$. It is easy to notice and prove the alternation of signs in the sequence $(x_n; y_n)$, that is, in the formula $R_{n+1} = R_{n-1} - R_n \cdot q_{n+1}$, the addition of either positive or negative numbers always occurs. This allows the "folding" algorithm by substituting the subtraction in the extended Euclidean algorithm for addition: $F_{n+1} = F_{n-1} + F_n \cdot q_{n+1}, F_n = (Q_n; P_n), F_{-2} = (1; 0), F_{-1} = (0; 1)$. The fractions P_n/Q_n are called convergents for the fraction a/b. The extended Euclidean algorithm in terms of convergents will look like this: $P_{n+1} = P_{n-1} + P_n \cdot q_{n+1}, Q_{n+1} = Q_{n-1} + Q_n \cdot q_{n+1}, P_{-2} = 0, P_{-1} = 1, Q_{-2} = 1, Q_{-1} = 0$. Let us prove that all convergents are irreducible. From the equality$a \cdot x_n + b \cdot y_n = r_n = d$, where $d = GCD(a; b)$, it follows that xn and yn are coprime, that is, the fraction P_n/Q_n is irreducible. But then the previous convergent P_{n-1}/Q_{n-1} will be irreducible, since it can be considered as the penultimate one in the decomposition of Pn / Qn into a continuous fraction. Thus, we have shown that the extended Euclidean algorithm can be considered as an algorithm for reducing fractions. As mentioned above, the signs xn and yn alternate, which can be written exactly as $x_n = (-1)^n \cdot Q_n,\ y_n = (-1)^{n+1} \cdot P_n$. We apply the extended Euclidean algorithm to P_{n+1}/Q_{n+1} : since P_n and Q_n are coprime, we obtain the equality $P_{n+1} \cdot x_n + Q_{n+1} \cdot y_n = 1$ or $P_{n+1} \cdot (-1)^n Qn + Q_{n+1} \cdot (-1)^{n+1} P_n = 1$, which is equivalent to $P_{n+1} \cdot Q_n - Q_{n+1} \cdot P_n = (-1)^n$, whence the formula for the difference of neighboring convergents is obtained $P_{n+1}/Q_{n+1} - P_n/Q_n = (-1)^n/(Q_n \cdot Q_{n+1})$. Other properties of convergents are obtained in the usual way from this formula. increases.

3. About the role of examples

Papert's book [3] draws attention to the term "personal thinking". We will interpret it as "relying on those ideas that the learner owns". Why does solving problems (not exercises) have such a positive effect on mathematical development?

Because this is a direct path to initiating the student's own judgments based on his OWN IDEAS. How to make the presentation understandable to everyone? Examples are a good tool for that. Firstly, they connect two different interpretations of a new idea (in this case, formal with a concrete one), and as you know, images based on internal connections are stored in memory. Secondly, they open the way for independent activity. Finally, and most importantly, they are a way of using the mechanism of internalization [4]. The independent "discovery" by the student of the various patterns outlined above can be supported by the structuring of his activities in the process of constructing and analyzing the protocols of the algorithm.

Conclusion

The report shows that when setting out the course of mathematics in technical universities it is advisable to focus on the algorithmic representation of the material. This will naturally connect the material with the activities of the programmer and thereby increase the applied character of the presentation. It is also shown that an analysis of algorithms can become an adequate replacement for the traditions of presenting material in a non-constructive style in the form of a series of theorems.

References

[1] Kudryavtsev T.V. *Psychology of technical thinking: the process and methods of solving technical problems.* - Moscow: Pedagogy, 1975 (rus)

[2] Krylov A.N. *The importance of mathematics for a shipbuilder. "Shipbuilding"* (No. 7 [43], July 1935) and in the "Bulletin of the USSR Academy of Sciences" (No. 7-8, 1938) (rus)

[3] Papert, S. (1996). *An Exploration in the Space of Mathematics Educations.* International Journal of Computers for Mathematical Learning, Vol. 1, No. 1, pp. 95-123, in 1996.

[4] Leontiev, A. N. *The Development of Mind*, a reproduction of the Progress Publishers 1981 edition, plus *"Activity and Consciousness"*, originally published by Progress Publishers, 1977, published by Erythrospress, see Erythrospress.com (1977)

Sergei Pozdniakov
Algorithmic Mathematics Department
Saint-Petersburg Electrotechnic University LETI
Saint-Petersburg, Russia
e-mail: pozdnkov@gmail.com

Elena Tolkacheva
Algorithmic Mathematics Department
Saint-Petersburg Electrotechnic University LETI
Saint-Petersburg, Russia
e-mail: eatolkacheva@etu.ru