FOLIA 273

Annales Universitatis Paedagogicae Cracoviensis

Studia ad Didacticam Mathematicae Pertinentia 10(2018)

ISSN 2080-9751
DOI 10.24917/20809751.10.9

Anna Pyzara

Algorithmisation as mathematical activity and skills in connection with mathematical modelling*

Abstract. Good preparation of students to the profession of teacher is very important. In my research I focus on improving the quality of teacher's preparation at the university level, and through it at school level in the future. I present the proposal of teaching mathematics with the use of algorithmisation. It is possible, because solutions to many mathematical problems can be expressed in the form of an algorithm. The process of algorithmisation of mathematical issues is associated with various types of mathematical activities. Therefore algorithmisation of mathematical problems forces the students to perform various mathematical activities.

This proposition deals with the problem of algorithmic computation of purely mathematical problems as well as those which deal with the problems of applying mathematics in everyday life. I consider algorithmisation as one of the forms of creating a mathematical model of a situation known from real world. Such interdisciplinary approach to teaching helps in developing students' skills better.

1. Introduction

Zofia Krygowska (1977a, 1977b) states that: "We consider development of mathematical activity of student as one of the most important goal of mathematics teaching." Inspired by this statement, I decided to explore issues of algorithmisation in didactics of mathematics, because this topic involves a whole series of mathematical activities and skills. It is good to remind what we understand by mathematical activity. According to Wanda Nowak (1989): "*Mathematical activity* of pupil is work of mind oriented on learning of concepts and on the mathematical type reasoning, which is stimulated by the situations that lead to formulation and

*2010 Mathematics Subject Classification: Primary: 97D50; Secondary: 97M10

Keywords and phrases: algorithmisation, mathematical modelling, quality of teaching, education

solving theoretical and practical problems." Therefore, my research concerns both purely mathematics issues and problems of application of mathematics in various areas of extramathematical human activity (it means, applying mathematics to solve problems from everyday life).

Why do I use algorithmisation when I explore teaching of mathematics? Because mathematics has two faces: conceptual and algorithmic. However, we cannot make a division into conceptual and algorithmic mathematics because these two aspects constantly permeate each other and they are equally important. In order to analyse conceptual elements, we need computational methods. On the other hand, algorithms treated separately from concepts are only automatic patterns for calculations. In my approach I do not deal with the problem of computability itself, which is a separate research area within mathematics.

2. Algorithm

There are many definitions of the algorithm, representing different levels of mathematical precision. Their review led me to accept the following definition:

"*Algorithm* is an ordered set of unambiguous, executable steps that determine a finite process, which leads to the realization of a certain task." (Brookshear, 2003)

- unambiguity
- effectiveness
- generality
- elementariness of operations.

Unambiguity means that it should precisely define the sequence of operations leading to the result. Therefore, a student who has mastered basic operations is able to get to the solution of a complicated task by doing the step-by-step activities planned in the scheme. The *effectiveness* of an algorithm guarantees that the resulting outcome is the correct solution of the task after a finite number of steps, whereas the *generality* condition means that an algorithm should comprise the whole class of tasks by working on parameters, the specification of which defines a given task. Of course, an algorithm always works in the same way for the same initial data. The feature of unambiguity imposes this.

The *elementariness of operations* means that each operation appearing in the scheme is controlled by the student. This feature is relative and depends on the skills of the student at a given level of education because some operations that are not elementary at a certain stage may become elementary in the further course of study (Krygowska, 1977a, 1977b).

Those features of algorithm force to specific working methods in teaching of mathematics, which has many advantages.

2.1. Algorithmisation in didactics of mathematics

Mathematics teaching with the use of algorithmization can be done in many ways, for example:

- teaching basic algorithms,
- execution of ready-made algorithms,
- comparing algorithms,
- algorithm analysis,
- algorithm supplementation.

One of them is also *creating algorithms* by students. This method requires from them variety mathematics activities, which are discussed in more detail in the next paragraph.

Application of algorithmisation in teaching brings many educational advantages, because the characteristics of the algorithm force the specific activity of the student. The didactic benefits of teaching mathematics with the use of algorithmisation are the following (Rams, 1982):

- knowledge of basic algorithms streamlines calculations, allows automatism,
- algorithm analysis allows you to see the precision and simplicity of a logical sequence of operations – method of operation,
- self-created algorithm:
  - requires logical thinking,
  - forces a very clear and unambiguous record solution plan,
  - develops reflective thinking – allows self-control;
- graphical writing allows:
  - non-verbal way of showing relationships and procedures,
  - comprehensive presentation of the method;
- algorithmisation enables systematization of knowledge and it is an IT-oriented teaching.

Those skills are reflected in the goals of mathematics education, included in the curriculum on every educational level, especially with regard to mathematical modelling. In this way mathematics, which is taught at school, forces the need to look at algorithms through the problem of mathematical modelling.

2.2. Algorithmisation as a form of mathematical modelling

Algorithmisation is one of the forms of mathematical modelling. What is mathematical modelling? According to Blum and Borromeo Ferri mathematical modelling is the process of translating between the real world and mathematics in both directions (Blum, Ferri, 2009). Mathematical model can be presented in the form of: formula, equation, equation system, function and **algorithm**.

Mathematical modelling is an important part of mathematics education, because one of the main objectives of this education is achieving by students skills to solve problems encountered in everyday life. Moreover, mathematical modelling is one of the main groups of skills included in the teaching objectives defined in Polish core curriculum of mathematics, and in similar documents all over the world.

Activities related to mathematical modelling are connected with the activities involved in creating the algorithm to solve the problem. That are (Blum, Ferri, 2009):

- constructing,
- simplifying,
- mathematizing (record of mathematical relations between considered variables),
- working mathematically,
- interpreting,
- validating.

These activities are important not only in mathematics, but also in the real world. In figure 1 we can see that mathematical modelling process is immersed in mathematics and in the real world. It is also important that it is a looping process. It is not enough to just do it once, but it should be repeated so as to verify and correct the first solution of the problem. It should be repeated to return to the mentioned activities.

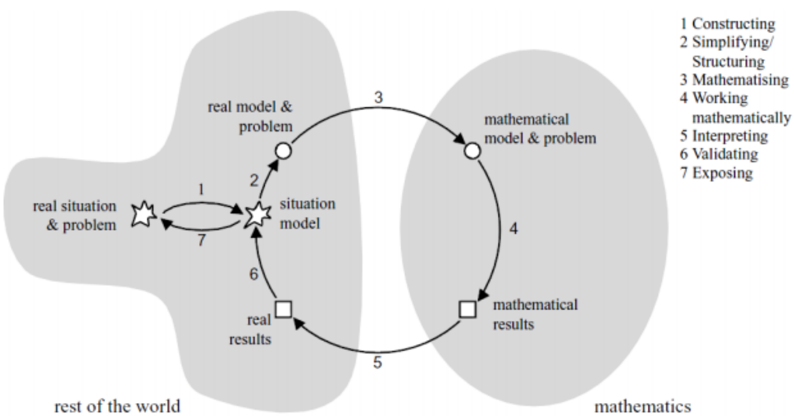

Fig. 1. Mathematical modelling process (Blum, Leiß, 2006)[1]

The issue of mathematical modelling is very extensive. In my research I confined myself to showing a mathematical model in the form of an algorithm. I am aware that this is strong narrowing of the topic. I do that, because in mathematics didactics there is no special place for algorithmisation and such an approach allows to use mathematical activities related to the algorithmisation in lessons of mathematics.

[1]The diagram is presented in the article: Mathematical Modelling: Can It Be Taught And Learnt? Blum, Ferri, 2009, p. 46.

3. Research – Methodology

3.1. Research questions

In my research, I tested the concept of teaching mathematics using mathematical modelling, in particular algorithmisation. I had a few courses with students of mathematics (future teachers) during which I checked their knowledge and skills related to mathematical modelling (in particular algorithmisation) and I tested whether introducing of this interdisciplinary teaching method would improve the students' mathematical competences.

I was looking for answer to the following research question: what changes in mathematical reasoning occur when future teachers work on developing algorithms for both pure and applied mathematical tasks?

3.2. Research group

The research group contains 16 university students of mathematics with specialization in teaching, future teachers of mathematics. I chose this group because they are the people who should theoretically have the knowledge and skills needed to solve the basic mathematical problems and problems encountered in everyday life that requires mathematical modelling. What is more, they should be familiar with the issues of algorithmisation during IT (information technology) classes.

3.3. Methodology

I made my research in 2016, but first there were several stages of initial research. Then I tested a variety of tasks, which allowed me to choose the types of research tasks, which I used in my research.

The research was based mainly on testing different activities, in which students created algorithms by themselves. I put a lot of emphasis on students' reflection, verification of their solutions and drawing conclusions concerning the correctness of their work.

The research consisted of three parts. In the first part, students created algorithms for solving problems based on their own knowledge and skills, which they gained thanks to their previous education. At the time students did an exercise 1 and exercise 3 (see: research tool). In the second part of the research I had classes with students during which they learned how to correctly create algorithms which help to solve problems and how to use mathematical modelling (particularly algorithmisation) in didactics of mathematics. The respondents built models together, analyzed their solutions, corrected mistakes. They compared different algorithms (solutions) and evaluated their correctness. Students were forced to reflect on the activities they performed because they had to write down the activities they had applied during the exercise. Then they got acquainted with the list of competences needed for modeling (see: Maaß, 2006). This list was the basis for creating by me a research tool, that made it possible to evaluate the correctness of algorithms. Following the list of competences, given by Maaß, I have distinguished a list of positive features that should include models created by students. When assessing students' answers, I gave the number of times a given feature appeared in the works (see: table 1). The students got to know the list of competencies so that they would be aware of which activities they need to train and what should be improved in their further work.

The third part of the research consisted in checking the effectiveness of the lessons. Students performed control exercises by themselves – they did, among other things, an exercise 2 and exercise 3. Comparison of the characteristics of the work done at the beginning and end of the course allowed for drawing some conclusions. This research lasted three weeks.

3.4. Research tool

I will discuss the research tool on four tasks, which represent two types of issues investigated in my research – purely mathematical tasks and problems from real world. During the research I used ten tasks, but due to the large amount of them I will limit this article to show four of them.

3.4.1. Problems from real world

Problems from real world force to create mathematical model of situation from reality. In such a case, it is important to:

- identify the factors that influence the considered problem,
- search and save relationships between them,
- verify the results obtained with regard to reality.

Exercise 1
Every year a school organizes a "Green trip". Create an algorithm that will allow to calculate the cost of participation of pupils in that tour when you know that:

- *the tour includes six nights;*
- *one adult can take care of a maximum of 15 pupils;*
- *arrival is by train;*
- *there is a visit to a local museum in the tour program.*

The algorithm should be so clear and understandable that it can be used by another organizer.

Exercise 2
Mr. Kowalski is a painter and he doesn't like to do calculations. Construct an algorithm for him that determines the amount of paint needed to paint room in cuboid shape.

Exercise 1 and 2 are examples of problem encountered in everyday life. These tasks require creating mathematical models of those situations. Students, which solve those problems, have to decide about detail level, decide which variables take to account and how formulate relationships between them. Without mathematization (which we understand as a record of problem in the form of mathematical relations that create a mathematical model of the considered situation), student has to verify correctness of algorithms with regards to reality.

3.5. Pure mathematic problems

Exercise 3 and 4 are examples of task from pure mathematics. They force to create an algorithm that solves square equations and square inequalities.

Exercise 3
Create an algorithm that solves square equations $ax^2 + bx + c = 0$.

Exercise 4
Create an algorithm that solves square inequalities $ax2 + bx + c > 0$.

These tasks require from students specific working methods. For example, purely mathematical tasks:

- require comprehensive understanding of methods, understanding of mathematical issues,
- require considering all possible solutions to the problem,
- verify the knowledge.

3.6. Example of correct solution of two tasks

The tasks presented in the previous paragraph represent two types of tasks. By showing example solutions I will present one task of each type (problem from real world and purely mathematical tasks).

We can see the example of correct solution of exercise 2 and exercise 4. The models presented here are my proposition to solve these tasks.

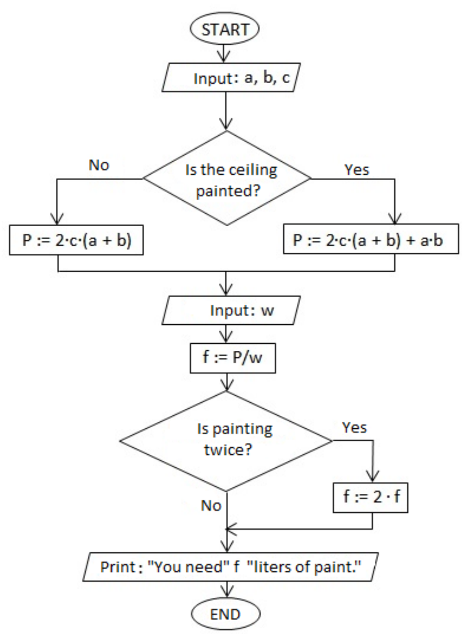

Fig. 2. Algorithm determining the amount of paint needed to paint the room (exercise 2)

I will present an example of an algorithm, that determines the amount of paint needed to paint the room in cuboid shape, in block diagram form. The algorithm loads information about the dimensions of the room and performance of the paint used, which determines the surface possible to paint with 1 liter of paint. I assume that the data for the room will be given in meters and the performance in m^2/l. Model, shown in the picture 2, allows the decision, whether the calculated amount of paint is also used to paint the ceiling. The possibility of double painting is also considered, if you paint the selected type of paint required. The algorithm produces the information that determines the number of liters of paint needed.

Algorithm specification:

Algorithm problem: Determining the amount of paint needed to paint a room in cuboid shape.

Input data: $a, b, c, w \in \mathbb{R}_+$,

Output data: the amount of paint – $f \in \mathbb{R}$, $f \geq 0$.

Working variables: $P \in \mathbb{R}_+$.

Designations of variables:
a *– length of the room,*
b *– width of the room,*
c *– height of the room,*
w *– paint performance, it means how many* m^2 *surface can be painted using* 1 *liter of paint,*
f *– the amount of liters of paint needed to paint the room.*

The algorithm presented here is quite general. At the same time this model is not too complicated. Larger details of this algorithm would allow, for example, the deduction surface of windows and doors or different colors (types) of paint. On the other hand, any additionally considered aspect of this situation would expand the algorithm, which would make it difficult to execute. Regardless of the level of detail, the prepared algorithm should be clear (unambiguity), general and effective – the proposed solution to the problem presented here meets these requirements.

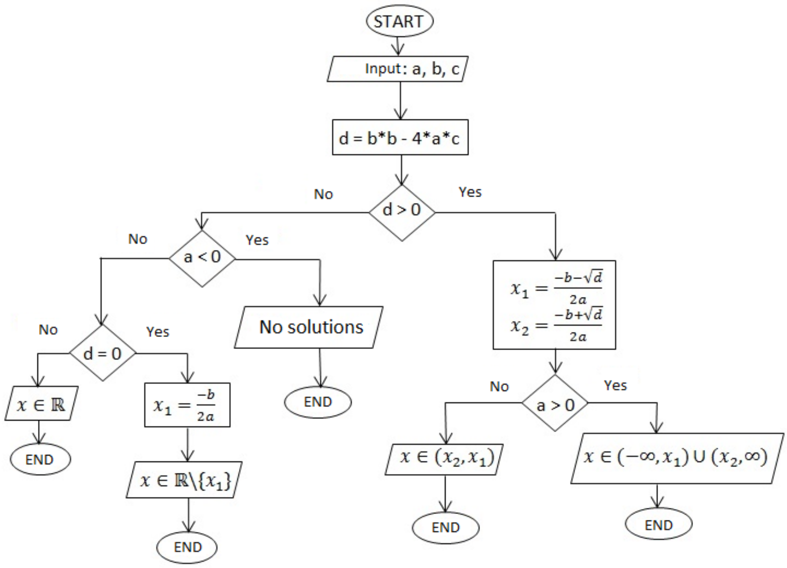

Fig. 3. Algorithm which solves square inequalities (exercise 4)

An exemplary algorithm for solving square inequality $ax^2 + bx + c > 0$ has been shown in block diagram form. It was assumed that the coefficient a is different from zero.

Therefore, the algorithm does not check the correctness of the data being loaded, because it was assumed that the data provided would meet the conditions specified in the specification of the algorithm.

Algorithm specification:

Algorithm problem: Solution of square inequalities $ax^2 + bx + c > 0$.

Input data: $a, b, c \in \mathbb{R}$, $a \neq 0$.

Output data: Set of solutions of inequality (interval, sum of interval, set of real numbers or set of real numbers without a single point) or information about the lack of solutions.

This algorithm considers all possible solutions of given square inequalities. The solution is given in the form of unambiguously defined intervals, and in the case of an empty set, it issues the information “No solutions”. The proposal is an optimal algorithm that preserves generality, uniqueness and effectiveness.

4. Initial student results – student errors

The mathematics students had many problems with constructing algorithms which solved the problems encountered in everyday life and the mathematical problems of everyday school life. Theoretically, students could solve tasks used in the research (as related to the problem familiar to them), but in spite of this they made many errors. Prospective teachers, despite the fact that they theoretically should have tested skills, can't apply them (especially in open situations). Students do not know the concept of mathematical modelling. They only have some of the competencies related to mathematical modelling. That is why (among other things) there are numerous errors in their works.

The most common mistakes are:

- they do not consider all possible solutions to the problem (in particular, they do not cover all solutions of the equation),
- they do not verify the obtained results,
- they use unauthorized constants in place of variables,
- they do not connect knowledge from different subjects, nor knowledge from everyday life.

A significant problem is the fact that students do not test, do not check correctness of the solution. Students do not verify the correctness of their solution – they focus on giving "result" fast. If there is some result, the activity ends.

No verification of the correctness of the solution implies another significant failure, namely students do not often consider all possible solutions to the problem. This demonstrates the misunderstanding of the fully discussed mathematical issues. For example, on figure 4 we can see that students do not consider a possibility of a negative parameter b (on the left side). What is more, the algorithm does not give information about what happens, when delta is less than zero.

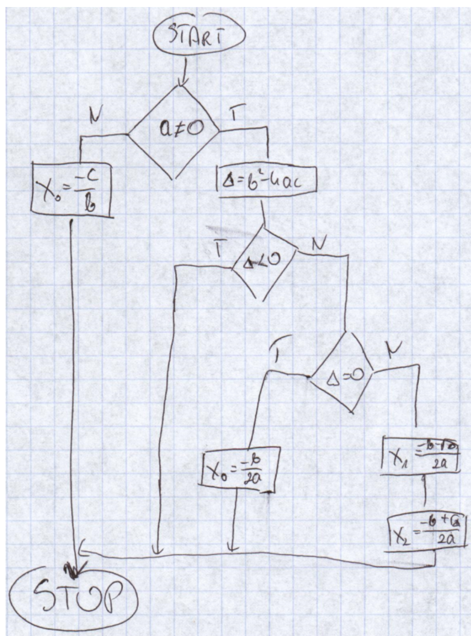

Fig. 4. Algorithm, which solves square equations

Another problem, which appears in students' works, especially when they were creating algorithm for everyday situation, is the use of unauthorized constants. Some students present the algorithm with specific data, they made some calculations, and that is all. They do not see differences between example and general method.

On figure 5 we can see algorithm which calculates cost of participated in school trip. There are only constant values, not variables. Such solution is an example calculation for a particular situation (specific case) – This is not an algorithm for solving a task that should be general.

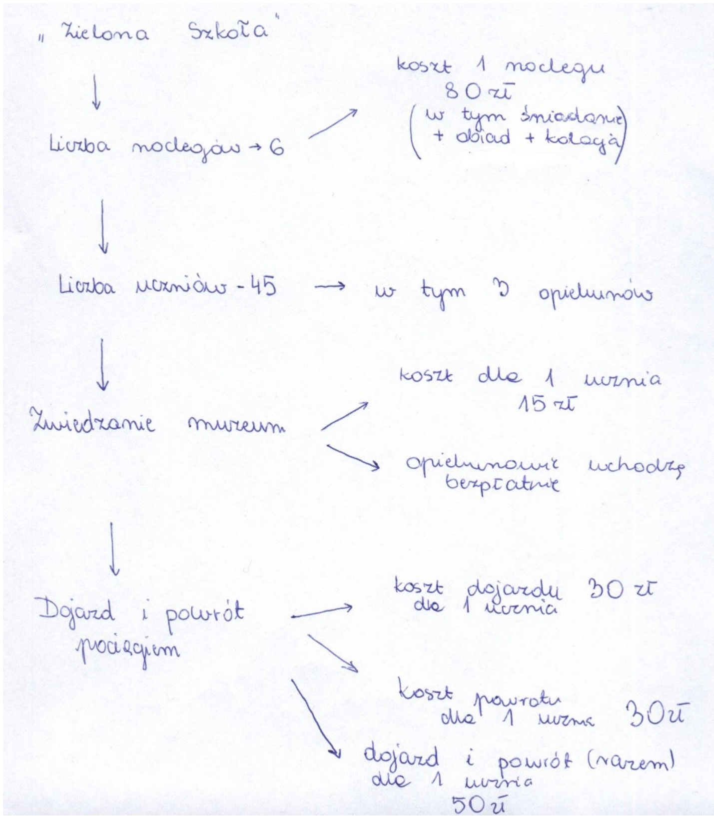

Fig. 5. Algorithm which calculates cost of participated in school trip

A big and common problem is that students do not connect knowledge from different subjects and also knowledge from everyday life. Many students think in the following way: "What is in school, that is in school; what is in real life, that is in real – we cannot connect them!". This clearly shows the work of students related to exercise 1 (school trip). In one of the student's work, the algorithm which calculates the cost of participation in school trip, contains information like: "We have to make a list of children, who will go on the trip.", "We have to choose teachers, who will go on the trip.", and so on. These are practical tips for organizing a school trip, however, they should not appear in the algorithm which calculates cost of the trip. This kind of mistakes students made mostly in tasks related to the situation in everyday life, however, they also occur in purely mathematical tasks.

The tasks considered in the study allow students to make transgression in their thinking, they show the possibilities and the need to connect knowledge from different areas.

In the student's works there were more detailed errors that are largely related to the correctness of representation of the algorithm, however, in this publication I will focus only on the main problems appearing in the student's works.

5. The results of the course

Observations collected during the first phase of the research allowed me to determine basic problems related to algorithmic skills. They also gave me the opportunity to test a variety of tasks that could be helpful in overcoming these problems. I used this experience by designing a course for academic students.

Students of mathematics (future teachers) took part in a short course. During this, students' skills related to mathematical modelling were developed (see: methodology), in particular related to the algorithm. The main emphasis was on the students' mathematical activities that were shown when building algorithms, which solve the problems in question.

Improving student skills can be easily seen when analyzing their work done at the end of the course. Figure 6 and 7 show exemplary solutions of the respondents.

In figure 6 we have example solution of exercise 2. There we have algorithm that solves Mr. Kowalski problem. A student who built this algorithm created mathematical model of a situation from reality. He mathematized it and verified the results with regard to reality. What is important, he used variables to mathematize it. The specification of the algorithm was shown in his work. Written notes were taken about the analysis of the problem carried out before construction of the mathematical model of the situation.

The presented solution is a properly functioning algorithm, which allows to solve the problem of Mr. Kowalski.

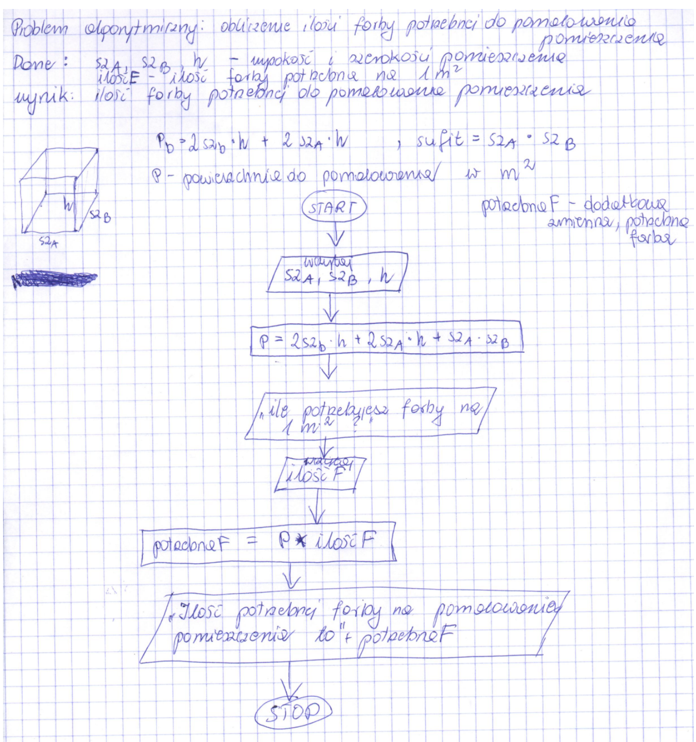

Fig. 6. Student's work – exercise 2

In figure 7 we can see example solution of exercise 4. There we have algorithm that solves square inequalities. We can see that student analyzed the situation. He considered all possible solutions and then created the algorithm. What is more, he used some example numbers to check correctness of the solution.

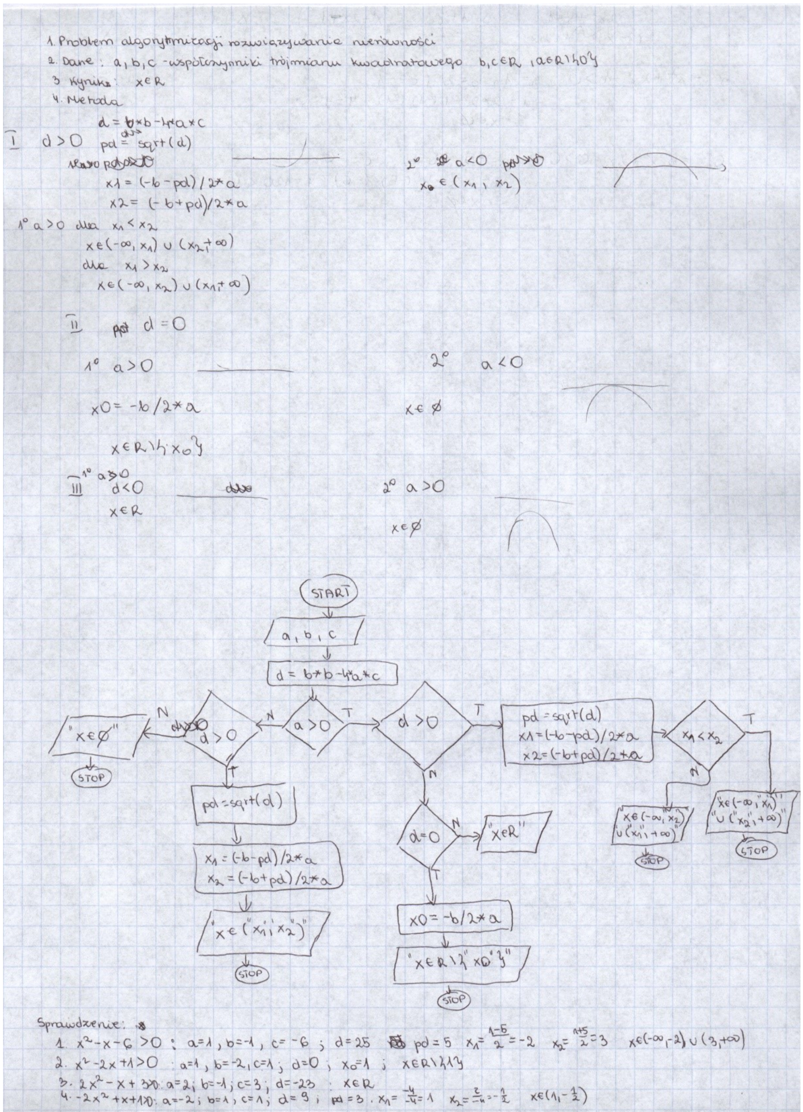

Fig. 7. Student's work – exercise 4

The characteristics of the approaches presented by participants in the study together with the number of occurrences are presented in table 1. The table shows a list of skills (a list of positive features of models created by students) that I have created by myself to evaluate students' competencies. The distinguished features of the solutions are strongly related to the competencies related to mathematical modeling. The numbers shown in the table indicate the number of occurrences of a given characteristic among 16 student works (the number of students who have applied a given property in their solution). The research analysis of students' solutions consisted in assessing each work for each property. Columns with exercise 1 and 3 show student results at the beginning of the course, and columns with exercise 2 and 4 at the end. Let me remind that ex. 1 and ex. 2 are problems from real world and ex. 3 and ex. 4 are pure mathematics problems. Comparing the results from exercise 1 to exercise 2, and the results from exercise 3 to exercise 4, we can see that after the course students improved their skills and their competencies. In most cases, there is a clear increase in the occurrence of the desired features, e.g. in most final works a description of the algorithm specification appeared.

Table 1. Characteristics of student work – students skills

Characteristics of student work	Ex. 1	**Ex. 2**	Ex. 3	**Ex. 4**
It's an algorithm specification	0	**13**	1	**12**
Written input	1	**14**	5	**13**
Data signs were given	12	**16**	5	**13**
All data signs were given	10	**9**	0	**3**
Specified data type	0	**9**	2	**9**
Specified all data type	0	**2**	2	**3**
The assumptions are written	8	**7**	4	**8**
The reflections are presented	3	**8**	2	**8**
There is a drawing illustrating the deliberations	0	**9**	0	**4**
Comments are added	5	**4**	1	**6**
The algorithm is tested	0	**1**	0	**4**
There are some calculations	0	**2**	0	**10**
The result is verified with regard to reality	0	**1**	–	–
The algorithm is improved (II version)	0	**1**	0	**4**
Everything is correct	1	**4**	1	**2**
Properly shown algorithm	1	**9**	3	**4**
Technically the algorithm works	5	**6**	4	**2**
Logically shown dependencies	6	**7**	7	**5**
Unauthorized constants were introduced	5	**4**	0	**0**
All possibilities were considered	–	–	**8**	7
Data is loaded	4	**16**	12	**16**
There is the start and the end of the algorithm	1	**14**	13	**16**
Conditional statements were used	8	**5**	14	**16**
Loops were used	0	**2**	0	**2**

The data presented in table 1 clearly show that the during the classes mathematical modelling competencies of the respondents significantly increased, especially algorithmic. Different skill areas have improved with varying degrees, however, a significant increase is noticeable. Regardless of whether we compare similarly themed tasks, or whether we take them all into consideration, in general, we notice a significant improvement in students' skills. Nevertheless, there are still areas that require further work, because such a short didactic block proved to be insufficient to fully master the desired skills (competences).

Comparing the performance of four tasks, it can be seen clearly, that the ability to save used data (variables) has improved significantly. In exercise 2 and 4, data signs were given 16 and 13 times, initially (in ex. 1 and 3) only 12 and 5 times respectively. Initially only 2 people specified data type, while 9 people did it by doing exercise 2, as well as doing 4. It is worth emphasizing that, at the end of the course, all students correctly loaded data into algorithms. Furthermore, students' competences in graphical representation of the algorithm have clearly improved. It should also be emphasized that the use of the form of the algorithm in solving tasks has contributed to make students use such elements as conditional statement and loop, which determine the approach to problem solving.

The obvious advantage of acquainting students with the mathematical modelling process (and with the creation of algorithms) is the fact, that students more often analyzed the problem before giving solutions – they wrote their reflections and sketched aids (drawings) to illustrate the observed problem (e.g. the beginning of the work shown in Figure 6). The reflections (testifying to the problem analysis) are presented 8 times in both task 2 and 4, while in tasks 1 and 3 they were 3 and 2 times respectively. In exercise 2 and 4 there was a drawing illustrating the deliberations 9 and 4 times, in the initial works no auxiliary drawings appeared. Previously students did not analyze the correctness of the algorithm – no correct version of the first model appeared in any work. As the result of such proceedings, there were numerous errors. It is worth emphasizing that as a result of the experience gained in the 10 solutions of task 4, calculations are used to control the algorithm.

A very important positive effect of the course is that there are works in which the presented algorithm is verified. In exercise 1 and 3 no one tested the solution presented by them by performing an algorithm for sample data, but in exercises 2 and 4 there are some calculations in over 10 works.

Compilation of data contained in Table 1 clearly show an increase in the competence of the students participating in the survey. Nevertheless, the compilation shows that some activities still need improvement. It is optimistic that even in these areas there is a slight improvement.

Conclusions

The studies discussed here have shown that many mathematical activities can be stimulated through the use of algorithmisation in teaching mathematics. Many problems (purely mathematical and those from everyday life) can be solved by building mathematical models of the situation. Showing models in the form of algorithms brings additional educational benefits. Preliminary studies have shown numerous deficiencies in future teachers' competences in the field of mathematical modelling. It is comforting that these competences can be complemented by the introduction of algorithmisation in the area of teaching of mathematics. Unfortunately, it turned out that conducting a short course was not sufficient to make up for any deficiencies. However, it is positive that in almost every area students' skills have increased (more or less).

In the summary, we can list several specific positive effects of the study:

- much improved data representation,
- students remembered to write the specification of the algorithm – they wrote the variables used and their markings,
- most people do not use unauthorized constant values,
- the form of the algorithm was improved visibly,
- students deal a bit better with extra-mathematical aspects of the tasks.

There are still areas of competence that still need improvement. Perhaps a longer course and more time spent on the algorithm would be better to reduce students' shortcomings. This aspect remains a question to be explored. However, there is a visible positive influence of the research on student's competences.

The main conclusions that can be given are as follows:

- Competences of students in mathematical modelling are insufficient – introductory classes should be introduced for their development.
- Participation of students in the discussed classes significantly increased their competence in mathematical modelling, especially in the algorithmisation.
- Algorithmisation can be taught and shown in a broader perspective.

At the end I would like to add that: pupils and students have different ways of perception – thus different learning needs. Variety of teaching methods make it easier to reach the individual needs of the students. My didactic proposal meets these needs. It allows a transgression in the approach to algorithmisation and its rediscovery in a broader perspective.

Finally, I post my message to the teachers: Let us not limit ourselves. Let us open up to interdisciplinary teaching methods. Let us take care of the overall development of the students!

References

Blum, W., Ferri, R.: 2009, Mathematical modelling: Can it be taught and learnt?, *Journal of Mathematical Modelling and Application* **1**(1), 45–58.

Brookshear, J.: 2003, *Informatyka w ogólnym zarysie*, WNT, Warszawa.

Krygowska, Z.: 1977a, *Zarys dydaktyki matematyki,* cz. 1, WSiP, Warszawa.

Krygowska, Z.: 1977b, *Zarys dydaktyki matematyki,* cz. 2, WSiP, Warszawa.

Krygowska, Z.: 1986, Elementy aktywności matematycznej, które powinny odgrywać znaczącą rolę w matematyce dla wszystkich, *Roczniki Polskiego Towarzystwa Matematycznego, Seria V, Dydaktyka Matematyki* **6**, 25–41.

Maaß, K.: 2006, What are modelling competencies?, *ZDM* **38**(2), 115–117.

Niss, M.: 2012, Models and modelling in mathematics education, *EMS Newsletter* **86**, 49–52.

Nowak, W.: 1989, *Konwersatorium z dydaktyki matematyki*, Państwowe Wydawnictwo Naukowe, Warszawa.

Perrenet, J., Zwaneveld, B.: 2012, The Many Faces of the Mathematical Modelling Cycle, *Journal of Mathematical Modelling and Application* **1**(6), 3–21.

Pyzara, A.: 2012, Algorithmisation in teaching mathematics, *Didactica Mathematicae* **34**, 51–68.

Pyzara, A.: 2014a, Modelowanie matematyczne sytuacji znanej z życia codziennego, *Współczesne Problemy Nauczania Matematyki* **5**, 233–255.

Pyzara, A.: 2014b, Przyszli nauczyciele wobec modelowania matematycznego, *Annales Universitatis Paedagogicae Cracoviensis Studia ad Didacticam Mathematicae Pertinentia* **6**, 121–136.

Pyzara, A.: 2014c, Creating an algorithm of a real-life situation as a form of mathematical modelling, *Didactics of Mathematics* **11**(15), 25–42.

Pyzara, A.: 2014d, Algorytmizacja jako umiejętność matematyczna, *Informatyka w Edukacji* 43–57.

Rams, T.: 1982, Problemy algorytmizacji, *Podstawowe zagadnienia z dydaktyki matematyki,* 119–141.

Sysło, M.: 1997, *Algorytmy*, WSiP, Warszawa.

Warwick, J.: 2007, Some reflections on the teaching of mathematical modeling, *The Mathematics Educator* **17**(1), 32–41.

Wójcicka, M.: 2005, *Wybrane metody i techniki aktywizujące. Zastosowanie w procesie nauczania i uczenia się matematyki*, Fraszka Edukacyjna, Warszawa.

Institute of Mathematics,
University of Maria Curie-Skłodowska
in Lublin
email: anna.pyzara@umcs.pl