MODELING AND ANALYSIS OF INFORMATION SYSTEMS, VOL. 28, NO. 1, 2021
journal homepage: www.mais-journal.ru

DISCRETE MATHEMATICS IN RELATION TO COMPUTER SCIENCE

Automated Teaching System “Sets” (Research for Organizing the 1st Part of the Project)

V. S. Rublev[1], M. D. Kondakov[1]

DOI: 10.18255/1818-1015-2021-1-90-103

[1]P. G. Demidov Yaroslavl State University, 14 Sovetskaya, Yaroslavl 150003, Russia.

MSC2020: 03B35, 03F03, 97E60
Research article
Full text in English

Received December 4, 2020
After revision December 14, 2020
Accepted March 12, 2021

The issues of building an automated learning system “Sets” which will allow students to master one of the important topics of the discipline “Discrete Mathematics” and to develop logical and mathematical thinking in this direction are studied. The corresponding topic of the 1st part of the project includes materials related to the concept of a set, operations on sets, algebra of sets, proofs of statements for sets, and the derivation of formulas for the number of set elements. The system is based on a construction of the statements proof editor for a set and of the formulas derivation editor for the number of set elements, both editors are to be used for teaching. The first of these allows students to split the original statement into a number of simpler statements, taken together equivalent to the original statement, to choose a method of proving each simple statement and to conduct their step-by-step proof. The second editor allows (using the inclusion-exclusion principle and the formula of the number of complement elements) to derive a step-by-step formula for the number of set elements through the specified numbers of elements for sets from which the resulting set is constructed.

An important part of the system is to monitor the correctness of all actions of students, and on this basis the entire learning system is developed. The logical supervision over the correctness of the selected action in the first editor is performed by a Boolean function created by the system and corresponding to this action and by checking it for identical truth. In the second editor, invariants such as characteristic strings of the set and of its number of elements are used for verification. The rest of the system is related to learning of set algebra and to preparation to editors usage. The main focus here is on the learning strategy in which testing the understanding of the learned material is rather rigorous and eliminating the random choice of answers. The division of the material into sections with verification of the success of teaching not only by tests, but also by exercises and tasks, allows students to master the complex logical and mathematical techniques of proving statements for sets and derivation of formulas for the number of set elements.

Keywords: computer learning, discrete sets, set algebra, statements for sets, step-by-step proof, number of set elements, formula derivation, correctness control

INFORMATION ABOUT THE AUTHORS

Vadim S Rublev correspondence author	orcid.org/0000-0002-0252-9958. E-mail: roublev@mail.ru PhD, Professor.
Maxim D Kondakov	orcid.org/0000-0002-7238-3675. E-mail: kondakovmd99@mail.ru student of the Department of Theoretical Informatics.

Funding: This work was supported by P. G. Demidov Yaroslavl State University Project № VIP-016.

For citation: V. S. Rublev and M. D. Kondakov, “Automated Teaching System “Sets” (Research for Organizing the 1st Part of the Project)”, *Modeling and analysis of information systems*, vol. 28, no. 1, pp. 90-103, 2021.

МОДЕЛИРОВАНИЕ И АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ, ТОМ 28, № 1, 2021
сайт журнала: www.mais-journal.ru

DISCRETE MATHEMATICS IN RELATION TO COMPUTER SCIENCE

Автоматизированная обучающая система «Множества» (исследования организации 1-й части проекта)

В. С. Рублев[1], М. Д. Кондаков[1]
DOI: 10.18255/1818-1015-2021-1-90-103

[1]Ярославский государственный университет им. П. Г. Демидова, ул. Советская, 14, г. Ярославль, 150003 Россия.

УДК 510.52:372.851
Научная статья
Полный текст на английском языке

Получена 4 декабря 2020 г.
После доработки 14 декабря 2020 г.
Принята к публикации 12 марта 2021 г.

Исследуются вопросы построения автоматизированной обучающей системы «Множества», которая позволит учащемуся освоить одну из важных тем дисциплины «Дискретная математика» и развить логико-математическое мышление в этом направлении. Соответствующая тема 1-й части проекта включает материал, связанный с понятием множества, операциями над множествами, алгеброй множеств, доказательствами утверждений для множеств, выводом формул для количества элементов множества. В основе системы лежит построение с целью использования для обучения редактора доказательства утверждений для множества и редактора вывода формул для количества элементов множества. Первый из них позволяет студенту разбить исходное утверждение на ряд более простых утверждений, в совокупности эквивалентных исходному утверждению, выбрать метод доказательства каждого простого утверждения и провести их пошаговое доказательство. Второй редактор позволяет, используя формулу включения и исключения и формулу количества элементов дополнения, вывести пошагово формулу для количества элементов множества через заданные количества элементов, связанных с ним множеств.
Важной частью системы является контроль правильности всех действий студента, и на этой основе разработана вся система обучения. Логический контроль правильности выбранного действия в первом редакторе осуществляется созданием системой булевой функции, соответствующей этому действию, и проверкой ее на тождественную истинность. Во втором редакторе для контроля используются такие инварианты, как характеристическая строка множества и характеристическая строка количества элементов множества. Остальная часть системы связана с обучением алгебре множеств и подготовке к использованию редакторов. При этом основное внимание уделяется стратегии обучения, при которой проверка понимания усвоенного материала является довольно строгой, исключающей случайный выбор ответов. Разбиение материала на секции с контролем успешности обучения не только тестами, но и упражнениями и задачами, позволяет студенту овладеть сложным логико-математическим аппаратом доказательства утверждений для множеств и вывода формулы для количества элементов множества.

Ключевые слова: компьютерное обучение, дискретные множества, алгебра множеств, утверждения для множеств, пошаговое доказательство, количество элементов множества, вывод формулы, контроль корректности

ИНФОРМАЦИЯ ОБ АВТОРАХ

Вадим Сергеевич Рублев автор для корреспонденции	orcid.org/0000-0002-0252-9958. E-mail: roublev@mail.ru кандидат физ.-мат. наук, профессор.
Максим Дмитриевич Кондаков	orcid.org/0000-0002-7238-3675. E-mail: kondakovmd99@mail.ru студент кафедры теоретической информатики.

Финансирование: Работа выполнена в рамках инициативной НИР ЯрГУ им. П. Г. Демидова № VIP-016.

Для цитирования: V. S. Rublev and M. D. Kondakov, “Automated Teaching System “Sets” (Research for Organizing the 1st Part of the Project)”, *Modeling and analysis of information systems*, vol. 28, no. 1, pp. 90-103, 2021.

Introduction

The challenge of teaching mathematical and computer sciences to the majority of students is caused by the low quality of school education, which is focused not on the thinking development, but on memorizing algorithms for solving some typical problems. This problem in the field of teaching students to develop new computer technologies can only be solved by individual teaching. But this approach requires a huge amount of additional time from the teacher. The solution is to develop computer-based automated teaching systems (ATS), which can be used not only to check knowledge, but also to conduct basic level teaching.

Most software systems called learning systems (for example, Moodle, Claroline, Dokeos, ATutor [1–11]) do not support the full learning cycle (methodologies) – they are just applications that provide access to texts, issue tests, and check student's memory ability. The best solution is to use various methods of adaptive learning in the learning processes, which are focused on a specific subject of study and on the individual characteristics of each student. Their features are:

- intelligent analysis of problem solutions;
- interactive support for problem solving;
- support in solving problems using examples;
- adaptive navigation support;
- adaptive view;
- adaptive support for collaboration of system users (students).

This provides the system with flexibility in relation to users and in relation to the presentation of a study material. In addition, the theoretical and methodological basis is the statement that teaching can be reduced to a set of the following components:

- information needed to study;
- control activities that allow you to test your knowledge of given materials;
- the way to assess the level of knowledge gained;
- follow-up management – an important and complex mechanism that makes the system namely teaching.

When developing such a system, you need to solve a number of problems: how to divide the material, how to check knowledge, and how flexible the system will be in relation to users.

The check by testing of student's memory is insufficient for the ATS for branches of exact sciences, which should teach data analysis, formalization, analysis, reasoning, and transformation. The use of computer algebra [12–15] is the basis for the construction of such systems. For example, one of the paper authors used this to construct ATS of computational complexity of algorithms [16–21].

In this paper, we consider computer-based learning models for proving statements for sets and deriving formulas for the number of set elements through specified numbers of other sets elements. These models can be divided into 2 groups: models for conducting of proofs for assertions or deriving of formulas for the number of set elements, and learning models that prepare students to use the first group of models.

1. The problem of constructing a proof editor of statements for sets

To solve the problem specified in the headline, select the following task sequence:

1. Limitations on the type of statements for sets for whose prove you need to build an editor.
2. Equivalent transformation of the sets included in the statement.
3. Splitting the basic statement into an equivalent set of simple statements.
4. The choice of the method of proving a simple statement and the selection of an initial set of premises in it.
5. Definition of the elementary step of the proof.
6. Control of the correctness proof by the editor.

1.1. Statement type constraints for sets

In the general case of the statement we will consider some *universal* set U and its subsets $X_1, X_2, \dots, X_n$. The statement uses formulas for these subsets, constructed with operations *complement, intersection, union*, and brackets, changing the order of these operations, if necessary.

We restrict ourselves to statements for sets of the following form

<*conjunction of set relations*> $\{\longrightarrow \mid \leftrightarrow\}$ <*conjunction of set relations*>.

So in example (1)

$$\overline{X_1 \cap \overline{X}_2 \cup \overline{X}_3} = (X_2 \cup X_3) \cap \overline{X}_1 \;\leftrightarrow\; X_1 \cap X_2 \cap X_3 = \varnothing \;\wedge\; X_2 \subseteq X_1 \cup X_3 \tag{1}$$

it is argued that the equality of two sets given by expressions takes place if and only if the intersection of the subsets is empty and the second subset is included in the union of the other two.

1.2. Equivalent transformation of sets included in the statement

To simplify the process of a proof conducted by a student, it is recommended to simplify the complex expressions of some sets of statements. In this case, we mean the representation of a complex expression in the form of a union of sets (and their intersections) in some cases or as the intersection of sets (and their unions) in other cases. So in example (1) transformation of the set on the left side of the set equality (it by A) denote

$$A \equiv \overline{X_1 \cap \overline{X}_2 \cup \overline{X}_3} = (\overline{X}_1 \cup X_2) \cap X_3 = (\overline{X}_1 \cap X_3) \cup (X_2 \cup X_3) \tag{2}$$

gives both views (2). The set of the right part of the equation (it by B) denote is already represented in (1) by an intersection, and the union is obtained by the following transformation

$$B \equiv (X_2 \cup X_3) \cap \overline{X}_1 = X_2 \cap \overline{X}_1 \cup X_3 \cap \overline{X}_1. \tag{3}$$

The reason why such representations of the set are important is the possibilities of simplifying the conduct of the proof. The representation of a set as a union of its subsets allows us to divide the further proof into separate branches, where the membership of an element to a set becomes easier by dividing it into cases of belonging to a particular subset, and the proof for each such branch is simplified and can be conducted separately. Representation in the form of intersection simplifies the proof of the conclusion about non-belonging to some set due to the fact that it is enough to obtain first the result about non-belonging to one of its subsets of intersection.

1.3. Splitting the basic statement into an equivalent set of simple statements

If there is an equivalent operation in the basic statement, the statement can be replaced by a conjunction of two statements with implications in one direction and the other. So the statement example (1) can be replaced by the conjunction of the following statements (with the replacement of the parts of the set equality by the introduced notations A and B):

$$A = B \;\longrightarrow\; X_1 \cap X_2 \cap X_3 = \varnothing \;\wedge\; X_2 \subseteq X_1 \cup X_3 \tag{4}$$

$$X_1 \cap X_2 \cap X_3 = \varnothing \;\wedge\; X_2 \subseteq X_1 \cup X_3 \;\longrightarrow\; A = B \tag{5}$$

Note that statement (4), having in conclusion the conjunction, can be spitted into 2 statements. In statement (5), the ratio of equality of 2 sets at the end of the implication can be replaced by the conjunction of 2 inclusion, and therefore the statement (5) can also be divided into 2. As a result, we obtain the following partition of the basic statement (1):

$$A = B \;\longrightarrow\; X_1 \cap X_2 \cap X_3 = \varnothing \tag{6}$$

$$A = B \longrightarrow X_2 \subseteq X_1 \cup X_3 \tag{7}$$

$$X_1 \cap X_2 \cap X_3 = \varnothing \wedge X_2 \subseteq X_1 \cup X_3 \longrightarrow A \subseteq B \tag{8}$$

$$X_1 \cap X_2 \cap X_3 = \varnothing \wedge X_2 \subseteq X_1 \cup X_3 \longrightarrow B \subseteq A \tag{9}$$

An example of reducing the proof of the basic statement (1) to the proof of 4 simple statements (6)-(9) shows that this can be done in other cases of the basic statements.

1.4. The choice of a method of proving a simple statement and the selection of the initial set of premises in it

In a simple statement, when performing the premises of the left part of the implication, it is necessary to prove the truth of the right part of the implication (call it a *target* statement). There are 2 methods of proof when the target is the ratio of inclusion of sets:

- **direct** method when we prove for an *arbitrary* element of the universal set, that from the belonging of an element to the included set follows its belonging to the including set (call it a *target output*), that is, the accuracy of the inclusion.
- **indirect** method, when we assume the opposite in the target statement (*exists an element of the universal set that belongs to the included set of the target relation of the inclusion of sets, but does not belong to the including set*) and by a consistent output come to a contradiction with the conjunction of premises left part of the implication to this simple statement.

For example, in simple statements (7)-(9) both methods are possible. However, for statement (7) it is more rational to apply the method (perhaps indirect fewer steps of proof), and for statements (8)and (9) the direct method is more rational.

If the target statement is the equality of a set to an empty set or to a universal set, then only the indirect method is rational. In this case, the existence of its element is opposite for an empty set, and the existence of an element that does not belong to this set is opposite for the equality of a certain set to a universal set. For a simple statement (6) you need to use an indirect method and show the contradiction with the premises of the statement.

The proof of any method begins with sequential steps, each of which is based on a premise. The initial premise in the direct method of proof is that an arbitrary element of the universal set belongs to the included set of the target statement. For example, for statement (8), the initial premise is the expression $\forall x : x \in A$.

The initial premise in the indirect method of proof is the denial of the target statement, which is expressed by the existence of an element that contradicts the target statement. For example, for (6), the initial statement premise is the expression $\exists x : x \in X_1 \cap X_2 \cap X_3$, and for (7) – statement $\exists x : x \in X_2, x \notin X_1 \cup X_3$.

In addition to the initial premise, conclusions can be based on assumptions statement related to the conditions (left side of the implication). The statement system prepares them as the initial set of premises, adding to it the initial premise. Meanwhile

1) to the equality of sets (for example, $C = D$) there correspond 4 premises (in the example, $x \in C \longrightarrow x \in D$, $x \in D \longrightarrow x \in C$, $x \notin C \longrightarrow x \notin D$, $x \notin D \longrightarrow x \notin C$);
2) to the inclusion case (for example, $X_2 \subseteq X_! \cup X_3$) there correspond 2 premises (in the example, $x \in X_2 \longrightarrow x \in X_1 \cup X_3$ and $x \notin X_1 \cup X_3 \longrightarrow x \notin X_2$);
3) to the equality of a set to the empty set (for example, $X_1 \cap X_2 \cap X_3 = \varnothing$) there correspond 1 premise (in the example, $x \notin X_1 \cap X_2 \cap X_3$);
4) and the equality of a set to the universal set (for example, $X_1 \cup X_2 \cup X_3 = U$) also matches 1 premise (in the example, $x \in X_1 \cup X_2 \cup X_3$).

For each elementary conclusion of the proof, if it is true, the system adds the conclusion as a premise to the set of premises of the proof or branch of the proof (more on that in the next section).

1.5. Definition of the elementary step of the proof

The proof is conducted with the help of a sequence of steps, at each of which an elementary conclusion is drawn, based on the indicated premises for conclusion. For educational purposes, we limit ourselves to only elementary conclusions based on no more than 2 premises. For elementary conclusions, the following ideas are used:

1. If an element belongs to a certain set (for example, premises $x \in C$), the output can be its belonging to any set, covering the set of the premises (in the example, the output $x \in C \cup D$).
2. If an element belongs to two sets (for example, 2 premises $x \in C$ and $x \in D$), the output can be an element belonging to the intersection of these sets (in the example output $x \in C \cap D$).
3. If an element belongs to a set (for example , $x \in C \cap D$), it belongs to each part of it (in the example, 2 outputs $x \in C$ and $x \in D$).
4. If an element belongs to a set (for example, the premise $x \in C$), it does not belong to its complement (in the example the output $x \notin \overline{C}$).
5. If an element does not belong to a set (for example, premise $x \notin C$), it belongs to its complement (in the example, the output $x \in \overline{C}$)).
6. If an element does not belong to two sets (for example, 2 premises $x \notin C$ and $x \notin D$), it does not belong to the union of these sets (in the example the output $x \notin C \cup D$).
7. If an element does not belong to a union of sets (for example, the premise $x \notin C \cup D$), it does not belong to any of these union sets (in the example, the output $x \notin C$ and the output $x \notin D$).
8. If an element belongs to a union of sets (for example, the premise $x \in C \cup D$), it can belong to one of these union sets (in the example, the output $x \in C$ and the output $x \in D$) – this output is called *the splitting of the cases*.

Note that the splitting into cases can be conducted in different ways. A partition where the sets of cases do not intersect is called *alternative*. As an example of the premise $x \in C \cup D$ you can write the following division into cases: the output $x \in C$ and the output $x \in D \cap \overline{C}$. This is especially important when for one case, the further conclusion is easily built. Then, for the second case, additional information is obtained, which can help in the further conclusion. If the first case is also difficult, it can be divided into 3 cases: the output $x \in C \cap \overline{D}$, the output $x \in C \cap D$ and the output $x \in D \cap \overline{C}$.

Note also that the division into alternative 2 cases of belonging to a certain or set non-belonging to this set can always be done without relying on premises (for example, the output $x \in C$ and the output $x \in \overline{C}$ form 2 cases and do not require a premise).

Each conclusion, if made correctly, is added as a premise to the preceding set of premises. But when cases appear, each of them is associated with its own independent branch of proof and many premises of this branch, which is formed from the previous set of premises by adding a new premise – the case output.

Each branch of the proof must end with either a white square denoting the receipt of the target output, or a black square denoting the receipt of a contradiction. If all branches of the proof from the opposite ended with a contradiction, the proof of the statement was successful. If in the direct method the proof of all branches ended, but there are branches that ended in success (a white square), the proof was successful.

1.6. Editor control of the proof

The system allows the trainee to build a proof of the basic statement. But the system has to control all his or her actions, starting from splitting into simple statements, selecting the method of proof with the organization of an initial premise, performing each elementary step of the proof up to the completion of each branch of the proof.

To this purpose, the system, at the end of the above actions (by pressing a button **Verify**), builds a Boolean function corresponding to a statement for sets (simple statements into which the basic statement is splitted, or statements of the original premise, or the statement of an elementary step, as implications of conjunctions of the premises and output of the elementary step or implication of the premise and disjunction of the output of cases) and verifies its identity truth. This Boolean function (let us call it *BIF* Boolean Identity Function) is constructed as follows:

1. Each set $x \in X_i$ is replaced with a boolean variable y_i, whose value is equal to the truth of the statement that the element x belongs to this set ($y_i \equiv x \in X_i$). Other sets are replaced in the same way. For example, the set A is replaced with a boolean variable a,whose name is a lowercase letter, corresponding to the name of the set and its true value coincides with the value of the statement of the belonging of an element x to this set, i.e. $a \equiv x \in A$.
2. Operations for sets of union, intersection and complement are replaced accordingly on operations of negation, conjunction and disjunction for the corresponding subsets of statements.
3. The equality for sets is replaced by the equivalence operation of the corresponding statements.
4. The inclusion for sets is replaced by the operation of implication of the corresponding statements.
5. The equality of the set to the empty set is replaced by the negation of the corresponding statement for the set.
6. The equality of the set to the universal set is replaced by the corresponding statement for the set.

So in the considered example (2) of identical representations of the set A we obtain the following BIF. $f_a = (\overline{y_1 \wedge \overline{y_2} \vee \overline{y_3}} \leftrightarrow (\overline{y_1} \vee y_2) \wedge y_3) \wedge (\overline{y_1 \wedge \overline{y_2} \vee \overline{y_3}} \leftrightarrow (\overline{y_1} \wedge y_3) \vee (y_2 \wedge y_3))$ and since it is identically true we, then by theorem *on the connection of expressions of the algebra of sets and the algebra of statements* and its consequences (see, for example [22] the system confirms the correctness of transformations of the set A.

Next, for a simple statement (6), the BIF will look like this: $f_6 \equiv (\overline{y_1 \wedge \overline{y_2} \vee \overline{y_3}} \leftrightarrow (y_2 \vee y_3) \wedge \overline{y_1}) \longrightarrow \overline{y_1 \wedge y_2 \wedge y_3}$. Its identical truth also confirms the correctness of the simple statement (6).

When conducting proof from the contrary of this statement, the student receives the initial premise. $\exists x : x \in X_1 \cap X_2 \cap X_3$ from the negation of the proved statement $\overline{X_1 \cap X_2 \cap X_3 = \varnothing}$ and BIF for verification gets the following expression: $f_{u1} \equiv y_1 \wedge y_2 \wedge y_3 \leftrightarrow \overline{\overline{y_1 \wedge y_2 \wedge y_3}}$, and since it is identically true, the initial premise is built correctly by the student.

From the received initial premise follows a conjunction of 3 elementary conclusions. $x \in X_1 \wedge x \in X_2 \wedge x \in X_3$, what is verified by the identical truth of BIF of the function of implication of the function of premise and conjunction of functions of elementary conclusions $y_1 \wedge y_2 \wedge y_3 \longrightarrow y_1 \wedge y_2 \wedge y_3$.

2. Formulas derivation editor for the number of set elements

The student's work with this editor can be divided into the following stages:

1. Derivation of the formula for the desired set determined by the problem through the original sets of the problem.
2. Derivation of the formula for the desired set determined by the problem through the original sets of the problem.
3. Derivation of the formula for the number of the desired set elements through the specified numbers of elements from other sets of the problem using the inclusion-exclusion principle, or using formulas for complements to the sets of the problem.

As an example, consider the following problem:

> *85% of citizens annually attend entertainment events: theaters, museums, cinemas. Moreover, 43% visit theaters, 25% visit museums, 7% visit cinemas and museums, 15% visit theaters and cinemas, and 5% visit all 3 shows, but 45% do not visit theaters or museums. What is the percentage of citizens who attend only 1 type of entertainment?*

First of all, students must enter the notations for the source sets and write a formula for the desired set. Let's introduce the notations T, M, and K, respectively, for the sets of citizens who visit theaters, museums, and cinemas.

A student can write a formula for the desired set based on the complement of the set of those who attend only one show:

$$X \equiv \overline{T \setminus (M \cup K) \cup M \setminus (T \cup K) \cup K \setminus (T \cup M)}, \tag{10}$$

but this student can write another formula based on taking union of those who do not attend anything and those who attend at least 2 types of shows:

$$X \equiv \overline{T \cup M \cup K} \cup T \cap M \cup T \cap K \cup M \cap K. \tag{11}$$

The system must check whether the entered formula matches the task conditions correctly. To do this we divide the universal set U into 8 pairwise disjoint subsets in the following order:

$$(\overline{T} \cup \overline{M} \cup \overline{K}, \overline{T} \cup \overline{M} \cup K, \overline{T} \cup M \cup \overline{K}, \overline{T} \cup M \cup K, T \cup \overline{M} \cup \overline{K}, T \cup \overline{M} \cup K, T \cup M \cup \overline{K}, T \cup M \cup K).$$

Let us assign each set of U to its characteristic string (CS) of 0,1, in which all its parts are defined as units. For example, for the set from the problem described above CS=(10010111). For sets T, M, and K, their CSs are respectively equal to (00001111), (00110011), (01010101). For the desired set X, its CS is obtained by applying to CSs of initial sets operations of bitwise conjunction, disjunction, and negation corresponding to the intersection, union, and complement operations included in the set formula. The ground for this follows, for example, from [22, 23]. Therefore, the correctness of any transformation of a set is controlled by the system by comparison with the CS of this set.

While deriving a formula for the number of elements, the inclusion-exclusion principle is used, as well as the formula for the number of elements for the complement set. The correctness of their application should also be monitored. To do this, the characteristic string of the quantity (CSQ) for the set elements which is similar to the CS is introduced. Its elements are equal to the multiplier with which the number of elements for the corresponding part of the set is included in the sum of the number of the set elements. While adding (or subtracting) such strings, their elements are added (subtracted) bitwise. Note that if for the number of set elements in CSQ all elements are from 0,1, then CSQ coincides with the CS of this set.

Let us illustrate this by the example under consideration. Firstly, let us introduce the numbers of elements for the sets given initially:

$$|T \cup M \cup K| = 85, |T| = 43, |M| = 25, |T \cap K| = 15, |M \cap K| = 7, |\overline{T} \cap \overline{M}|.$$

Let us take a simpler formula (11) for determining the numbers of elements of the set X. It is a union of 2 disjoints sets: complement to the union of main sets X1, X2, X3, and the union of pairwise intersections of these sets. Therefore, using the inclusion-exclusion principle, we get:

$$|X| = |\overline{T \cup M \cup K}| + |T \cap M \cup T \cap K \cup M \cap K|. \tag{12}$$

The first term in (12), using the number of elements of the complement, is replaced by the difference of sets:

$$|\overline{T \cup M \cup K}| = |U| - |T \cup M \cup K|, \tag{13}$$

and the second one is expanded by the inclusion-exclusion principle

$$|T \cap M \cup T \cap K \cup M \cap K| = |T \cap M| + |T \cap K| + |M \cap K| - 2|T \cap M \cap K|. \tag{14}$$

Checking the CSQ for (14) also gives a valid equality

$$(00010111) = (00000011) + (00000101) + (00010001) - 2(00000001).$$

Using the equalities (13) and (14), we make a substitution in (12)

$$|X| = |U| - |T \cup M \cup K| + |T \cap M| + |T \cap K| + |M \cap K| - 2|T \cap M \cap K|. \tag{15}$$

In the resulting formula (15), all the terms are known except . Let us express the formula for it using the complement

$$|T \cap M| = |U| - |\overline{T \cap M}| = |U| - |\overline{T} \cup \overline{M}|. \tag{16}$$

Transform the subtractive to (16) using the inclusion-exclusion principle and then, using the formula for complements, we obtain

$$|\overline{T} \cup \overline{M}| = |\overline{T}| + |\overline{M}| - |\overline{T} \cap \overline{M}| = 2|U| - |T| - |M| - |\overline{T} \cap \overline{M}|. \tag{17}$$

Substitute the result in formula (16) and then after substituting in (15) we get the final formula (18)

$$|X| = |T| + |M| + |\overline{T} \cap \overline{M}| + |T \cap K| + |M \cap K| - 2|T \cap M \cap K| - |T \cup M \cup K|. \tag{18}$$

Its check by performing actions on the CSQ of terms

$$(00001111) + (00110011) + (11000000) + (00000101) + (00010001) - (00000002) - (01111111) =$$
$$= (11121224) - (01111113) = (10010111)$$

shows a coincidence with the CSQ of the desired set. Substituting the values of summands in formula (18)

$$|X| = 43 + 25 + 45 + 15 + 7 - 2 \cdot 5 - 85 = 40. \tag{19}$$

We get the answer to the question of the problem: *40% of citizens, except for those who attend only 1 type of entertainment.*

3. Checking by the formula derivation correctness editor

The example above for solving a problem with supervising shows how the editor can check the logic of the formula derivation. But this check is not yet complete, because, firstly, the check of syntactic correctness of the entered expressions is not defined, and, secondly, the exact steps of the formula derivation are not defined. For entering expressions, it is needed to use syntax verification using the finite state grammar. For step-by-step derivation, it is needed to introduce a system of elementary transformations of formulas for a set and for the number of its elements. In this case, an elementary transformation is performed for the selected fragment of the transformed formula.

For sets formulas, we include the following elementary transformations in the system:

1. Removing the double complement.
2. Permutation of sets in the intersection.
3. Permutation of the sets in the union.
4. Absorption for intersection-repeats of sets in an intersection or an intersection with the universal set are removed.
5. Absorption for union-repeats of sets in a union or a union with an empty set are removed.
6. Removal of parentheses-usage of the distributive law for intersection with the union of sets.
7. Factorization using the distributive law for unions of intersections.
8. Transformation of the difference, that is the change of the subtraction to the intersection with the complement.
9. Removing the complementation sign-using the duality law.
10. Inclusion of the complementation sign-using the duality law.

11. Notation for a part of the set-introduction of a new formula.
12. Formula substitution-replacing the notation of a set by its formula.
13. Replacing of a union with a complement with the universal set.
14. Replacing of the universal set by a union of an arbitrary set with its complement.
15. Replacing of an intersection with a complement with an empty set.
16. Replacing of an empty set by an intersection of an arbitrary set with its complement.

Using elementary set transformations can lead to a long derivation. Therefore, along with the list of elementary transformations, we introduce a list of grouped transformations:

1. Multiple operation application-if there are several fragments selected in the formula, the elementary operation is applied to each fragment;
2. Multiple absorption-in the selected fragment of an intersection or a union of sets, repetitions of sets are removed, as well as unions with an empty set or an intersection with the universal set.

When we derive the formula for the number of set elements, we use the following transformations:

1. The inclusion-exclusion principle for a union of a given number of sets;
2. Replacing the number of elements in a difference of sets by the difference of their elements numbers-if the subtracted set is included in the reduced set;
3. Replacing the number of set elements with the difference between the quantities of the universal set and the complement of a set;
4. Mutual deletion of similar terms with different signs;
5. Deleting of terms with empty intersections;
6. Collecting terms-checks the similarity of all selected terms and the correctness of the coefficient calculation.

4. Teaching models

4.1. Teaching strategy

All teaching material can be divided into two large categories. The first category includes sections that provide basic theoretical knowledge and concepts on set definition, operations with sets, and set algebra. The main purpose of these sections is to develop and train a student’s memory using memorization skills. These sections are most often checked by testing.

However, the course program also includes teaching the informal usage of mathematical techniques, and at this point there are additional difficulties. These include technical difficulties associated with entering of formulas and their transformation, and difficulties associated with an insufficient level of the logical thinking of a student. This student must achieve the goal by breaking a derivation or a proof into parts, choosing methods for these parts, and constructing a sequence of derivations leading to the goal. Thus, defined is the second category of sections related to the mathematical methodology of conducting a proof of an assertion for sets or deriving a formula for the number of set elements.

The following features distinguish the second category of sections from the first one. Firstly, the check of the material of sections from the second category cannot be based only on testing, because it is necessary to check the student’s ability to use various mathematical techniques. Secondly, when studying the material, it is necessary to teach the student to link individual techniques into a purposeful process by constructing a sequence of studied techniques. Thirdly, you need to check the ability to link several processes when solving the final problem. One of the tools used in checking the sections material from the second category is the algorithm for checking the syntactic correctness of the input and the algorithm for checking the correctness of the derivation.

The material and the check interact with the third component of the system, which is responsible for determining the volume of the material in a single session, and sets a set of tasks and their number, as well as other system parameters. It is namely this component that gives flexibility to the system and distinguishes

ATS from an application that can only output text and a set of tests based on it. Further we give a description of the proposed scenario for the interaction between the system and a user.

After logging in, the user sees a list of course sections that are divided into accessible and yet not accessible ones according to the course plan. Sections must be completed one after the other in their order (the principle of linear learning). Upon the entry to the section, the student is provided with its material for studying. After studying the material, the student proceeds to the check of the acquired knowledge, for which he or she must complete the initial number of tasks defined in the section (tests, exercises, tasks). Each question in the test offers several possible answers (usually 6), randomly selected from a pre-defined set of answers for each task. Among the suggested answers, there may be several correct answers, or all correct answers, or none correct answers. The student must mark all correct answers. But if, in his opinion, there are no correct answers, then he should choose this answer. The system considers the task completed if the student has marked all the correct answers and none of the incorrect ones. Otherwise, it displays one of the texts: ”Not all correct answers are marked”, ”Some of the marked answers are not correct”, or a combination of them, and provides a text fragment of the section material associated with the error. After studying this fragment or re-studying the entire material of the section, the student has the opportunity to re-answer the question of the task. If the user does not complete the task again, it is replaced with two additional tasks. Thus, the number of tasks to complete the section may increase. The session with the student is terminated when a certain number of tasks in the section is reached, and the student is able to return to learning only after a certain break. If a student reaches a session interruption several times in a row, his or her account in the system is temporarily blocked, and he or she is called to the teacher. In the case when the student first earned a lot of additional tasks, and then began to answer correctly, the number of control tasks begins to decrease in some progression. This approach makes the student carefully and thoughtfully treat the material of the section. In this way, not only the check of the section knowledge is organized, but also the learning. Only after completing all the tasks the student is able to proceed to the study of the material in the next section.

However, there are sections whose material is based on the previous sections and can not be mastered without absolute possession of the material of these previous sections. In such sections, an additional check of the knowledge of previous sections is carried out. In this case, the number of initial tasks for repeating the section is reduced to one.

Note also that custom teaching parameters are defined for configuring ATS. For example, the initial number of test tasks in each section or sub-section, the number of tasks to be added if the answer is incorrect, the number of tasks to be reduced for every 2 consecutive correct answers, and so on.

4.2. Splitting into sections

All the material is divided into the following sections and subsections.

1. Set definition. Tests to determine correct or incorrect answers.
2. Relations of sets:
   (a) The equality relation. Tests to determine correct or incorrect answers.
   (b) The inclusion relation. Tests to determine correct or incorrect answers.
3. Operations on sets:
   (a) Sets union. Tests to determine correct or incorrect answers.
   Proof and transformation exercises.
   (b) Intersection and complement of sets. Tests to determine correct or incorrect answers. Proof and transformation exercises.
4. Set algebra. Tests to determine correct or incorrect answers.
   Exercises on specification of sets formulas.
5. Methodology of proving statements for sets.

   (a) Premises and a conclusion of a proof step. Tests to determine correct or incorrect answers. Exercises for drawing conclusions from given premises. Exercises for determining premises for a given conclusion.
   (b) Splitting into cases. Tests to determine whether the answers are correct or incorrect in relation to the branches of a proof. Exercises for splitting into normal and alternative cases.
   (c) Direct method of proof. Tests to determine whether the answers are correct or incorrect regarding the completion of proof branches. Exercises on determination of the initial premise and the conclusion derived from it.
   (d) Indirect method of proof. The tests to determine correct and incorrect answers relative to the branches of a proof and their completion. Exercises for setting the initial premise. Exercises for double division into cases.
6. Errors in a proof. Tests to determine correct and incorrect answers.
7. System of transformations for formulas for a set.
   (a) Elementary transformations. Tests to determine correct and incorrect answers.
      i. Permutations and absorptions. Exercises on transformations.
      ii. Transformations with parentheses. Exercises on transformations.
      iii. Transformations with complement sets. Exercises on transformations.
      iv. Notation and substitution of formulas. Exercises on transformations.
   (b) Multiple transformations.
      i. Multiple application of an operation. Exercises on transformations.
      ii. Multiple absorption. Exercises on transformations.
8. Final section for assertion proof. The problem of proving a statement for sets.
   (a) Numerical illustrations with general cases when all conditions of a statement are met and when each of the conditions is not met.
   (b) Venn Diagrams with general cases of fulfillment of all statement conditions and non-fulfillment of each of the conditions.
9. Formulas connecting the numbers of sets elements.
   (a) The inclusion-exclusion principle. Tests to determine correct and incorrect answers. Exercises for derivations of a formula for sets with singularities.
   (b) Formula for the number of elements for the difference of sets and the complement of a set. Tests to determine correct and incorrect answers.
10. System of transformations for formulas for the number of set element. Tests to determine correct or incorrect answers.
   (a) The inclusion-exclusion principle for a union of a given number of sets. Exercises on transformations.
   (b) Replacing the number of elements in sets difference by the difference of their elements numbers. Exercises on transformations.
   (c) Replacing the number of set elements by the difference between the quantities of the universal set and the set complement. Exercises on transformations.
   (d) Mutual deletion of similar terms with different signs. Exercises on transformations.
   (e) Deleting terms with empty intersections. Exercises on transformations.
   (f) Collecting similar terms. Exercises on transformations.
   (g) Notation and substitution of formulas. Exercises on transformations.
11. Final section for formulas for the number of set elements. Task to derive a formula for the number of set elements.

Conclusion

We hope that the described approach to construct an automated teaching system for proving statements for sets will be implemented and will show effectiveness in teaching this subject and developing students' logical and mathematical thinking.

References

[1] A. Büchner, *Moodle 3 Administration. Third Edition.* Packt Publishing, 2016.

[2] S. K. Udaya and T. V. Vamsi Krishna, "E-Learning: Technological Development in Teaching for school kids", *International Journal of Computer Science and Information Technologies*, vol. 5, no. 5, pp. 6124–6126, 2014.

[3] M. Kerres, D. Meister, F. von Gross, and U. Sander, *Enzyklopädie Erziehungswissenschaft.* 2012.

[4] S. Clark and J. Baggaley, "Assistive Software for Disabled Learners", *The International Review of Research in Open and Distributed Learning*, vol. 5, no. 3, 2004. [Online]. Available: https://doi.org/10.19173/irrodl.v5i3.198.

[5] A. Nagy, "The Impact of E-Learning", in *E-Content: Technologies and Perspectives for the European Market*, Berlin: Springer-Verlag, 2005, pp. 79–96. [Online]. Available: https://doi.org/10.1007/3-540-26387-X_4.

[6] I. E. Allen and J. Seaman, *Changing Course: Ten Years of Tracking Online Education in the United States.* Babson Survey Research Group, 2013.

[7] D. Burgos *et al.*, Eds., *Higher Education Learning Methodologies and Technologies Online*, Springer, 2019, pp. 6–7.

[8] A. Holzinger *et al.*, Eds., *Machine Learning and Knowledge Extraction.* Springer, 2019.

[9] Y. Zhang and D. Cristol, *Handbook of Mobile Teaching and Learning.* Springer, 2019.

[10] S. Kong and H. Abelson, *Computational Thinking Education.* Springer Nature, 2019.

[11] G. Hanna, D. Reid, and M. de Villiers, *Proof Technology in Mathematics Research and Teaching.* Springer, 2019.

[12] M. England and other, Eds., *Computer Algebra in Scientific Computing*, Springer, 2019.

[13] J. H. Davenport, Y. Siret, and E. Tournier, *Computer algebra: systems and algorithms for algebraic computation.* Academic Press, 1988.

[14] J. Gathen and J. Gerhard, *Modern Computer Algebra.* Cambridge University Press, 2013.

[15] V. B. Taranchuk, *The main functions of computer algebra systems*, in Russian. Minsk: BSU, 2013.

[16] A. V. Ermilova and V. S. Rublev, "Problems of mathematical thinking development for students basing on example of teaching system to the course "Algorithms and computational complexity analysis"", in Russian, in *Proceedings of IX International theoretical and practical conference Information Technologies and IT-Education*, Moscow: INTUIT.RU, 2014, pp. 297–304.

[17] V. S. Rublev and M. T. Yusufov, "Automated system for teaching computational complexity analysis of algorithms", in Russian, *Modern Information Technologies and IT-Education*, vol. 12, no. 1, pp. 135–145, 2016.

[18] V. S. Rublev and M. T. Yusufov, "Automated teaching system "Analysis of algorithms computational complexity" (research for organizing the 1st part of the project)", in Russian, *Modern Information Technologies and IT-Education*, vol. 13, no. 2, pp. 170–178, 2017.

[19] V. S. Rublev and M. T. Yusufov, “Automated System for Teaching Computational Complexity of Algorithms Course”, in Russian, *Modeling and Analysis of Information Systems*, vol. 24, no. 4, pp. 481–495, 2017.

[20] V. S. Rublev and M. T. Yusufov, “Models for teaching analysis of algorithm computational complexity”, in Russian, in *3rd International on Computer Algebra and Information Technologies*, Odessa: ONSU, 2018, pp. 157–160.

[21] V. S. Rublev and D. R. Vakhmyanin, “Automated system for teaching «Proof of statements for sets»”, in Russian, *Modern Information Technologies and IT-Education*, vol. 15, no. 4, pp. 1014–1027, 2019.

[22] V. S. Rublev, *Boolean functions (individual work #4 and #5 for the course “Discrete Mathematics”)*, in Russian. Yaroslavl: YSU, 2018.

[23] V. S. Rublev, *Sets (individual work #1 for the course “Discrete Mathematics”)*, in Russian. Yaroslavl: YSU, 2018.