

УДК 510.52:372.851

DOI: 10.25559/SITITO.15.201904.866-875

Автоматизированная обучающая система «Доказательство утверждений для множеств»

В. С. Рублев*, Д. Р. Вахмянин

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

* roublev@mail.ru

Аннотация

Исследуются вопросы построения автоматизированной обучающей системы «Доказательство утверждений для множеств», которая позволит учащемуся освоить одну из первых тем дисциплины «Дискретная математика» и развить логико-математическое мышление в этом направлении. В основе системы лежит построение редактора утверждений для множеств, позволяющего обучаемому разбить исходное утверждение на ряд более простых утверждений в совокупности эквивалентных исходному утверждению, выбрать метод доказательства каждого простого утверждения и провести их пошаговое доказательство.

Важной частью системы является контроль правильности всех действий обучаемого, и на этой основе разработана вся система обучения. Разработанная автоматная грамматика позволяет проверить правильность вводимого текста пользователем при выборе соответствующего действия. Логический контроль правильности выбранных действий в редакторе (разбиение исходного утверждения, выбор начальной посылки метода доказательства, ввод шага доказательства) контролируются созданием системой булевой функции, соответствующей этому действию, и проверкой ее на тождественную истинность.

Остальная часть системы связана с обучением алгебре множеств и подготовке к использованию этого редактора. 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 контролем не только тестами, но и упражнениями и задачами, позволяет обучаемому овладеть сложным логико-математическим аппаратом доказательства утверждений для множеств.

Ключевые слова: компьютерное обучение, множества, алгебра множеств, доказательство утверждений для множеств, пошаговое доказательство, контроль корректности.

Для цитирования: Рублев В. С., Вахмянин Д. Р. Автоматизированная обучающая система «Доказательство утверждений для множеств» // Современные информационные технологии и ИТ-образование. 2019. Т. 15, № 4. С. 866-875. DOI: 10.25559/SITITO.15.201904.866-875

© Рублев В. С., Вахмянин Д. Р., 2019



Контент доступен под лицензией Creative Commons Attribution 4.0 License.
The content is available under Creative Commons Attribution 4.0 License.



Automated Training System “Proof of Statements for the Set”

V. S. Rublev^{*}, D. R. Vahmyanin

¹ P.G. Demidov Yaroslavl State University, Yaroslavl, Russia

14 Sovetskaya Str., Yaroslavl 150003, Russia

*roublev@mail.ru

Abstract

The issues of construction of automated training system “Proof of statements for sets” are investigated, which will allow the student to master one of the first topics of discipline “Discrete mathematics” and develop logical and mathematical thinking in this direction. The system is based on the construction of a statement editor for sets, allowing the trainee to split the original claim into a series of simpler statements in a combination of equivalent to the original claim, chooses a method of proving each simple claim, and conduct their step-by-step proof.

An important part of the system is to control the correctness of all actions of the trainee, and on this basis the whole training system is developed. The developed automatic grammar allows checking the correctness of the entered text by the user when selecting the appropriate action. Logical verification of the correctness of selected actions in the editor (splitting of the original statement, selection of the initial message of the proof method, entry of the proof step) is controlled by creation of a Boolean function corresponding to this action by the system and check for its identical truth.

The rest of the system is related to teaching set algebra and preparing to use this editor. An important part of the system is the control of the correctness of all actions of the trainee, and on this basis the training system has been developed. This focuses on a learning strategy in which testing the understanding of learned material is fairly rigorous, avoiding random choice of answers. Dividing the material into sections with control of the success of training by control not only tests, but also exercises and tasks, allows the trainee to master a complex logical and mathematical apparatus of proof of statements for sets.

Keywords: computer training, sets, sets algebra, proving statements for sets, editor for teaching the proof, step by step proof, control of step correctness.

For citation: Rublev V.S., Vahmyanin D.R. Automated Training System “Proof of Statements for the Set”. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technologies and IT-Education. 2019; 15(4):866-875. DOI: 10.25559/SITITO.15.201904.866-875



Введение

В настоящий период проблемы обучения основной массы студентов математическим и компьютерным наукам связаны с неразвитостью мышления, вызванного низким качеством школьного образования. Значительная масса выпускников школ не умеют читать (понимать прочитанное), не умеют рассуждать, поскольку их этому недостаточно обучали. Эти проблемы могут быть решены только индивидуальным обучением, но такой подход требует от преподавателя огромных дополнительных временных затрат, превышающих в несколько раз часы, выделяемые учебным планом. Поэтому напрашивается путь решения в разработке компьютерных автоматизированных систем обучения (АСО), с помощью которых можно не только контролировать знания, но вести обучение. Существует большое количество программных систем, называемых обучающими¹ (например, Moodle, Claroline, Dokeos, ATutor [1-13]), но большинство из них не поддерживают полный цикл обучения (методики) – это всего лишь приложения, которые предоставляют доступ к текстам, выдают задания и проверяют их [2]. Более продвинутым решением является использование в процессе обучения различных методик, которые изменяют поведение системы по отношению к конкретному пользователю в зависимости от его индивидуальных особенностей. Для решения этой задачи предлагается использовать адаптивные обучающие системы (АОС), основная парадигма которых – адаптация к каждому студенту. Адаптивные технологии обучения появились сравнительно недавно, но завоевали популярность среди разработчиков обучающих систем. В основе адаптивного обучения лежат следующие методологии:

- построение последовательности курса обучения;
- интеллектуальный анализ решений задач;
- интерактивная поддержка в решении задач;
- поддержка в решении задач на примерах;
- адаптивная поддержка в навигации;
- адаптивное представление;
- адаптивная поддержка сотрудничества пользователей системы (обучающихся).

Применение этих методологий обеспечивает систему гибкостью в отношении к пользователю и в отношении к представлению материала для изучения.

В дополнение к концепции адаптивных систем *теоретической и методологической базой* становится утверждение о том, что обучение может быть сведено к совокупности следующих составляющих:

- информации, необходимой для изучения;
- контрольным мероприятиям, позволяющим проверять знания по данным материалам;
- способу оценки уровня полученных знаний;
- последующему управлению – важному и самому сложному механизму, делающему систему именно обучающей.

Следовательно, необходимо ответить на следующие вопросы: как провести разбиение, как вести контроль и насколько гибкой будет система по отношению к пользователю.

В основном многие компьютерные системы осуществляют контроль тестированием памяти студента, а потому могут обучать некоторым определениям, но не их использованию. АСО

для разделов точных наук должно обучать анализу данных, их формализации, анализу, рассуждениям и преобразованиям. Использование компьютерной алгебры [14-17] лежит в основе построения таких систем. Например, один из авторов данной работы использовал это для построения АСО вычислительной сложности алгоритмов [18-20].

В данной работе рассматриваются модели компьютерного обучения доказательству утверждений для множеств. Эти модели можно разделить на 2 группы: модели проведения доказательства утверждения для множеств и модели обучения, подготавливающие студента к использованию моделей первой группы. Модели первой группы воплощены в редакторе доказательства утверждений для множеств.

Редактор доказательств утверждений для множеств

Для построения редактора выделим следующую последовательность задач:

1. Ограничения вида утверждений для множеств, для доказательства которых нужно построить редактор.
2. Эквивалентное преобразование множеств, входящих в утверждение.
3. Разбиение исходного утверждения на эквивалентную совокупность простых утверждений.
4. Выбор метода доказательства простого утверждения и выделение начального множества посылок в нем.
5. Определение элементарного шага доказательства.
6. Контроль редактором корректности доказательства.
7. Контроль редактором синтаксической правильности ввода.

Ограничения вида утверждений для множеств

В общем случае утверждения мы будем рассматривать некоторое универсальное множество U и его подмножества X_1, X_2, \dots, X_n . В утверждениях используются формулы для этих подмножеств, конструируемые при помощи операций *дополнения, пересечения, объединения*, а также скобок, изменяющих при необходимости порядок выполнения этих операций. Мы ограничимся утверждениями для множеств следующего вида импликации или эквиваленции двух конъюнкций отношений множеств:

<Конъюнкция отношений множеств> $\{ \rightarrow \mid \leftrightarrow \}$ <Конъюнкция отношений множеств.>

Так в примере (1)

$$\overline{X_1 \cap \overline{X_2} \cup \overline{X_3}} \cap \overline{X_1} \leftrightarrow X_1 \cap X_2 \cap X_3 = \emptyset \wedge X_2 \subseteq X_1 \cup X_3 \quad (1)$$

утверждается, что равенство двух множеств, заданных выражениями, имеет место тогда и только тогда, когда пересечение подмножества пусто и второе подмножество включено в объединение двух других.

Эквивалентное преобразование множеств, входящих в утверждение

Для упрощения процесса доказательства, проводимого обучаемым, рекомендуется ему упростить сложные выражения некоторых множеств утверждения. При этом имеется в виду

¹ Claroline [Электронный ресурс]. URL: <https://claroline.net/> (дата обращения: 10.08.2019).



представление сложного выражения в виде объединения множеств (или их пересечений) в одних случаях или в виде пересечения множеств (и их объединений) в других случаях.

Так для примера (1) преобразование множества левой части равенства множеств (обозначим его через A)

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

дает оба представления (2). Множество правой части равенства (обозначим его через B) в (1) уже представлено пересечением, а объединение получается следующим преобразованием

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

Причина, по которой такие представления множества могут быть важны, являются возможности упрощения ведения доказательства. Так представление множества в виде объединения его подмножеств позволяет разделить дальнейшее доказательство на отдельные ветви, где принадлежность элемента множеству становится проще разбиением на случаи принадлежности его тому или иному подмножеству, а доказательство для каждой такой ветви упрощается и может вестись отдельно. Представление в виде пересечения упрощает доказательство вывода о непринадлежности какому-то множеству за счет того, что достаточно получить сначала результат о непринадлежности одному из его подмножеств пересечения.

Разбиение исходного утверждения на эквивалентную совокупность простых утверждений

Если в исходном утверждении есть операция эквиваленции, то утверждение можно заменить конъюнкцией 2 утверждений с импликациями в одну и другую сторону. Так пример утверждения (1) можно заменить конъюнкцией следующих утверждений (с заменой частей отношения равенства множеств на введенные обозначения A и B):

$$A = B \rightarrow X_1 \cap X_2 \cap X_3 = \emptyset \wedge X_2 \subseteq X_1 \cup X_3, \quad (4)$$

$$X_1 \cap X_2 \cap X_3 = \emptyset \wedge X_2 \subseteq X_1 \cup X_3 \rightarrow A = B. \quad (5)$$

Заметим, что утверждение (4), имеющее в заключении конъюнкцию, можно разбить на 2 утверждения. В утверждении (5) отношение равенства 2 множеств в заключении импликации можно заменить на конъюнкцию 2 отношений включения, а потому утверждение (5) также можно разбить на 2. В результате получим следующее разбиение исходного утверждения (1):

$$A = B \rightarrow X_1 \cap X_2 \cap X_3 = \emptyset \quad (6)$$

$$A = B \rightarrow X_2 \subseteq X_1 \cup X_3 \quad (7)$$

$$X_1 \cap X_2 \cap X_3 = \emptyset \wedge X_2 \subseteq X_1 \cup X_3 \rightarrow A \subseteq B \quad (8)$$

$$X_1 \cap X_2 \cap X_3 = \emptyset \wedge X_2 \subseteq X_1 \cup X_3 \rightarrow B \subseteq A \quad (9)$$

Пример сведения доказательства исходного утверждения (1) к доказательству 4 простых утверждений (6)-(9) показывает, что подобным образом можно поступать и в других случаях исходных утверждений.

Выбор метода доказательства простого утверждения и выделение начального множества посылок в нем

В простом утверждении при выполнении посылок левой части импликации необходимо доказать истинность правой части импликации (назовем ее *целевым* утверждением). Существуют 2 метода доказательства, когда целевым является отношение включения множеств:

прямой метод, когда мы для *произвольного* элемента универсального множества доказываем, что из принадлежности элемента включаемому множеству следует его принадлежность

включающему множеству (назовем это *целевым выводом*), то есть верность отношения включения заключения;

косвенный метод, когда мы предполагаем противное в целевом утверждении (*существует элемент универсального множества, который принадлежит включаемому множеству целевого отношения включения множеств, но не принадлежит включающему множеству этого отношения*) и последовательным выводом приходим к противоречию с конъюнкцией посылок левой части импликации для этого простого утверждения.

Так, например, в простых утверждениях (7)-(9) возможны применения обоих методов. Однако для утверждения (7) более рационально применять косвенный метод (возможно меньшее количество шагов доказательства), а для утверждений (8) и (9) более рационален прямой метод.

Если же целевым утверждением является равенство некоторого множества пустому множеству или универсальному множеству, то рационален только косвенный метод. В этом случае противным для пустого множества является существование его элемента, а противным для равенства некоторого множества универсальному множеству является существование элемента, который не принадлежит этому *некоторому* множеству. Так для простого утверждения (6) нужно применить косвенный метод и показать выводом противоречие с посылками условия утверждения.

Доказательство любого метода начинается с последовательных шагов вывода, каждый из которых опирается на посылки. Начальной посылкой при прямом методе доказательства является принадлежность произвольного элемента универсального множества включенному множеству целевого утверждения. Например, для утверждения (8) начальной посылкой является выражение $\forall x : x \in A$.

Начальной посылкой при косвенном методе доказательства является отрицание целевого утверждения, которое выражается в существовании элемента, который противоречит целевому утверждению. Например, для утверждения (6) начальной посылкой является выражение $\exists x : x \in X_1 \cap X_2 \cap X_3$, а для утверждения (7) — $\exists x : x \in X_2, x \notin X_1 \cup X_3$.

Помимо начальной посылки выводы могут основываться на посылках, связанных с условиями утверждения (левой части импликации утверждения). Система их оформляет как начальное множество посылок, добавляя к нему начальную посылку. При этом равенству множеств (например, $C = D$) соответствуют 4 посылки (в примере $x \in C \rightarrow x \in D, x \in D \rightarrow x \in C, x \notin C \rightarrow x \notin D, x \in D \rightarrow x \notin C$); отношению включения (например, $X_2 \subseteq X_1 \cup X_3$) соответствуют 2 посылки (в примере $x \in X_2 \rightarrow x \in X_1 \cup X_3$ и $x \notin X_1 \cup X_3 \rightarrow x \notin X_2$);

равенству множества пустому (например, $X_1 \cap X_2 \cap X_3 = \emptyset$) соответствует 1 посылка (в примере $x \notin X_1 \cap X_2 \cap X_3$); равенству универсальному множеству (например, $X_1 \cup X_2 \cup X_3 = U$) также соответствует 1 посылка (в примере, $x \in X_1 \cup X_2 \cup X_3$).

При каждом элементарном выводе доказательства, если он верный, система добавляет этот вывод в качестве посылки в множество посылок доказательства или ветви доказательства (об этом в следующем разделе).

Определение элементарного шага доказательства

Доказательство проводится при помощи последовательности



шагов, на каждом из которых делается элементарный вывод, на основе указанных посылок для вывода. При этом мы в учебных целях ограничимся только элементарными выводами, основанными на не более чем 2 посылках. Для элементарных выводов используются следующие идеи:

1. Если элемент принадлежит некоторому множеству (например, посылка $x \in C$), то выводом может являться его принадлежность любому множеству, охватывающему множество посылки (в примере, вывод $x \in C \cup D$);
2. Если элемент принадлежит двум множествам (например, 2 посылки $x \in C$ и $x \in D$), то выводом может являться принадлежность элемента пересечению этих множеств (в примере вывод $x \in C \cap D$);
3. Если элемент принадлежит множеству (например, $x \in C \cap D$), то он принадлежит каждой его части (в примере, 2 вывода $x \in C$ и $x \in D$);
4. Если элемент принадлежит множеству (например, посылка $x \in C$), то он не принадлежит его дополнению (в примере вывод $x \notin \bar{C}$);
5. Если элемент не принадлежит множеству (например, посылка $x \notin C$), то он принадлежит его дополнению (в примере вывод $x \in \bar{C}$);
6. Если элемент не принадлежит двум множествам (например, 2 посылки $x \notin C$ и $x \notin D$), то он не принадлежит объединению этих множеств (в примере вывод $x \notin C \cup D$);
7. Если элемент не принадлежит объединению множеств (например, посылка $x \notin C \cup D$), то он не принадлежит любому из этих множеств объединения (в примере, вывод $x \notin C$ и вывод $x \notin D$);
8. Если элемент принадлежит объединению множеств (например, посылка $x \in C \cup D$), то он может принадлежать одному из этих множеств объединения (в примере, вывод $x \in C$ и вывод $x \in D$) — такой вывод называют *разбиением на случаи*.

Заметим, что разбиение на случаи можно проводить различным образом. Разбиение, когда множества, относящиеся к каждому из случаев, не пересекаются, называется *альтернативным*. Так для примера посылки $x \in C \cup D$ можно написать следующее разбиение на случаи: вывод $x \in C$ и вывод $x \in D \cap \bar{C}$. Это особенно важно в том случае, когда для одного случая дальнейший вывод легко выстраивается. Тогда для второго случая получается дополнительная информация, которая может помочь в этом случае дальнейшему выводу. Если же первый случай также труден, то можно разбить на 3 случая: вывод $x \in C \cap \bar{D}$, вывод $x \in C \cap D$ и вывод $x \in D \cap \bar{C}$.

Отметим также, что разбиение на альтернативные 2 случая принадлежности некоторому множеству и непринадлежности этому множеству можно всегда сделать, не опираясь на посылки (например, вывод $x \in C$ и вывод $x \notin \bar{C}$ образуют 2 случая и не требуют посылок).

Каждый вывод, если он сделан корректно, добавляется в качестве посылки в предшествующее множество посылок. Но при появлении случаев с каждым из них связывается своя независимая ветвь доказательства и множество посылок этой ветви, которое образуется из предшествующего множества посылок добавлением новой посылки — вывода случая.

Каждая ветвь доказательства должна заканчиваться либо белым квадратом, обозначающим получение целевого вывода (т. е. что и требовалось доказать), либо черным квадратом, обо-

значающим получение противоречия. Если все ветви доказательства от противного закончились противоречием, то доказательство утверждения прошло успешно. Если в прямом методе доказательство всех ветвей закончилось, но есть ветви, закончившиеся успехом (белым квадратом), то доказательство этого утверждения закончилось успешно.

Контроль редактором корректности доказательства

Система позволяет обучаемому строить доказательство исходного утверждения. Но система должна контролировать все его действия, начиная от разбиения на простые утверждения, выбора метода доказательства с организацией начальной посылки, выполнение каждого элементарного шага доказательства вплоть до завершения каждой ветви доказательства. Контроль можно разделить на 2 части:

- **контроль синтаксиса** вводимых выражений;
- **контроль корректности доказательства** синтаксически верного ввода.

С этой целью система при завершении обучаемым каждого вышеописанного действия (нажатием например кнопки **Проверить**) осуществляет такой контроль. В этом разделе мы опишем проверку корректности доказательства, а в следующем разделе опишем синтаксический контроль.

Для контроля корректности доказательства система в соответствии с совершаемым действием строит *булеву функцию*, соответствующую утверждению для множеств:

- **простых утверждений**, на которое разбито исходное утверждение;
- **утверждения начальной посылки** для простого утверждения;
- **утверждения элементарного шага**, как импликации конъюнкции посылок и вывода элементарного шага или импликации посылки и дизъюнкции выводов разбиения на случаи.

Затем система проверяет эту булеву функцию на тождественную истинность. Эта булева функция (назовем ее *БФТИ*) строится следующим образом:

1. Каждое утверждение $x \in X_i$ заменяется на высказывание $y_i \equiv x \in X_i$, а утверждения принадлежности множеству (например, $x \in A$) заменяется на высказывание, обозначаемое строчной буквой названия множества (в примере $a \equiv x \in A$);
2. Операции для множеств дополнения, пересечения и объединения заменяются соответственно на операции отрицания, конъюнкции и дизъюнкции для соответствующих подмножеств высказываний;
3. Отношение равенства для множеств заменяется операцией эквиваленции соответствующих высказываний;
4. Отношение включения для множеств заменяется операцией импликации соответствующих высказываний;
5. Равенство множества пустому множеству заменяется отрицанием соответствующего высказывания для множества.
6. Равенство множества универсальному множеству заменяется соответствующим высказыванием для множества.

Так в рассмотренном примере (2) тождественных представлений множества A мы получаем следующую БФТИ
 $f_a = (y_1 \wedge y_2 \vee y_3) \leftrightarrow ((y_1 \vee y_2) \wedge y_3) \wedge (y_1 \wedge y_2 \vee y_3) \leftrightarrow (y_1 \wedge y_3) \vee (y_2 \wedge y_3)$ и так как она тождественно истинная мы, то по теореме о связи выражений алгебры множеств и алге-



бры высказываний и ее следствиям (см., например, [5]) система подтверждает правильность преобразований множества A . Далее, для простого утверждения (6) БФТИ будет выглядеть следующим образом:

$f_6 \equiv (y_1 \wedge y_2 \vee y_3 \leftrightarrow (y_2 \vee y_3) \wedge y_1) \rightarrow y_1 \wedge y_2 \wedge y_3$. Ее тождественная истинность также подтверждает правильность полученного простого утверждения (6).

При проведении доказательства от противного этого утверждения обучаемый получает начальную посылку $\exists x : x \in X_1 \cap X_2 \cap X_3$ из отрицания доказываемого утвержде-

ния $\overline{X_1 \cap X_2 \cap X_3} = \emptyset$ и БФТИ для проверки получает следующее выражение $f_{u1} \equiv y_1 \wedge y_2 \wedge y_3 \leftrightarrow y_1 \wedge y_2 \wedge y_3$, и так как она тождественно истинна, то начальная посылка построена обучаемым верно.

Из полученной начальной посылки следует конъюнкция 3 элементарных выводов $x \in X_1 \wedge x \in X_2 \wedge x \in X_3$, что проверяется тождественной истинностью БФТИ функции импликации функции посылки и конъюнкций функций элементарных выводов $y_1 \wedge y_2 \wedge y_3 \rightarrow y_1 \wedge y_2 \wedge y_3$.

Контроль редактором корректности ввода выражений

Для контроля корректности ввода строится следующая автоматная грамматика доказательства утверждений для множеств.

<знак конъюнкции> ::= ^

<знак импликации> ::= \rightarrow

<знак эквиваленции> ::= \leftrightarrow

<знак операции> ::= $\bar{\quad} \mid \cap \mid \cup$

<знак отношения элемента и множества> ::= $\in \mid \notin$

<знак отношения множеств> ::= $= \mid \subseteq$

<множество> ::= <set> _____

<set> ::= $\emptyset \mid \langle \text{set} \rangle \mid \langle \text{set} \rangle \{ \{ X_1 \mid X_2 \mid X_3 \} \{ \cap \mid \cup \} \langle \text{set} \rangle \}$

<исходная посылка> ::= <set> <знак отношения множеств> <set>

<начальная посылка> ::= <начальная посылка прямого метода доказательства> | <начальная посылка косвенного метода доказательства>

<начальная посылка прямого метода доказательства> ::= $\forall x \in \langle \text{set} \rangle$

<начальная посылка косвенного метода доказательства> ::= $\exists x \notin \langle \text{set} \rangle, x \in \langle \text{set} \rangle$

<утверждение> ::= x <знак отношения элемента и множества> <set>

<отношение множеств> ::= <set> <знак отношения множеств> <set>

<конъюнкция отношений множеств> ::= <отношение множеств> |

<конъюнкция отношений множеств> [<знак операции конъюнкции> <отношение множеств>]

<исходное утверждение для множеств> ::=

<посылка> ::= <утверждение> | <начальная посылка> | <исходная посылка>

<вывод> ::= <утверждение>

<конец ветви доказательства> ::= <знак «что и требовалось доказать»> | <знак противоречия>

<знак «что и требовалось доказать»> ::= \square

<знак противоречия> ::= \blacksquare

<отношение множеств> ::= <set> <знак отношения множеств> <set>

<конъюнкция отношений множеств> ::= <отношение множеств> |

<конъюнкция отношений множеств> [\wedge <отношение множеств>]

<исходное утверждение для множеств> ::=

<конъюнкция отношений множеств> { \rightarrow | \leftrightarrow } <конъюнкция отношений множеств>

<простое утверждение для множеств> ::=

<конъюнкция отношений множеств> \rightarrow <отношение множеств>



Примечания:

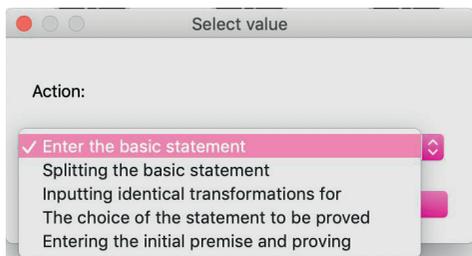
1. Названия в угольковых скобках называются нетерминальными символами в отличие от терминальных символов знаков, букв, цифр, запятых и т.д.
2. Знак ::= означает «есть» в определении для нетерминального символа слева от знака. Справа дается определение этого символа через другие символы.
3. Знак | означает «или» в определении.
4. Выражение в фигурных скобках означает выбор одного из элементов в них.
5. Выражение в квадратных скобках означает необязательный элемент.

Проверка синтаксиса выражения начинается с проверки правильности скобок. Для этого в строке символов последовательно считаются скобки: каждая открывающая скобка увеличивает счетчик на 1, каждая закрывающая скобка уменьшает счетчик на 1. Счетчик во время счета не должен принимать отрицательные значения и в конце счета должен быть равен 0 (как и перед началом счета). Каждой открывающей скобке с некоторым номером счетчика соответствует ближайшая закрывающая скобка с тем же номером счетчика перед ней.

При проверке синтаксиса выражения последовательно слева направо в строке символов проверяются нетерминальные символы, входящие в определение этого выражения. При проверке нетерминального символа определяется первый символ в проверяемой части строки и по нему выбирается из альтернативных определений, разделенных знаком | (или), соответствующая часть определения. Если ее нет, то выражение ошибочно. Все пробелы при проверке игнорируются.

Интерфейс редактора

Интерфейс редактора представляет собой конструктор. Используя диалоговое окно (рис.1), обучаемый выбирает необходимые ему элементы для работы. С помощью панели инструментов (рис. 2) он вводит данные в главном окне программы (рис. 3).



Р и с. 1. Диалоговое окно

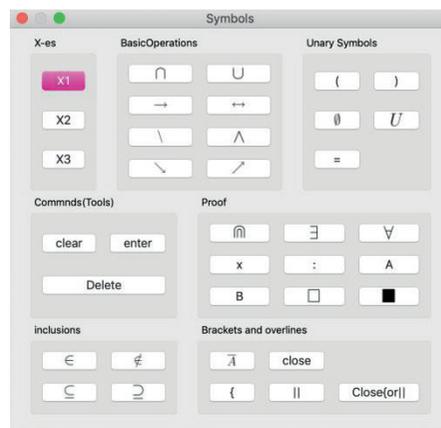
F i g. 1. Dialog window

Процесс доказательства делится на несколько шагов:

1. Сначала при запуске, вызывается диалоговое окно с выбором действий (рис. 1), где обучаемый выбирает одно из них для дальнейшей работы.
2. Выбор *Ввод исходного утверждения* создает окно ввода.
3. Выбор *Ввод тождественных преобразований для некоторых множеств* (им даются имена А и В) создает 2 поля ввода с кнопками *Проверка преобразований*.
4. Выбор *Разбиение исходного утверждения* создает номер разбиения исходного утверждения.
5. Выбор *Доказываемое утверждение* вызывает всплывающее окно, в котором отображаются ранее введенные раз-

биения исходного утверждения.

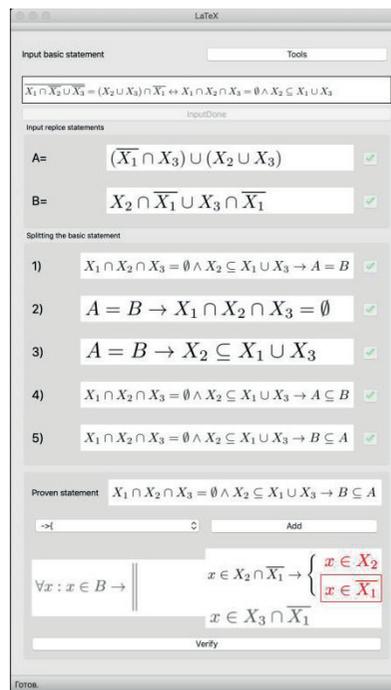
6. Выбор *Ввод начальной посылки шага доказательства* создает кнопку **Проверка** и поле ввода.



Р и с. 2. Панель инструментов

F i g. 2. Toolbar

После ввода текста элементарного вывода или начальной посылки доказательства выводится одна или несколько кнопок в виде красных квадратов, с нажатием на одну из которых образуется поле ввода текста очередного элементарного вывода. Выбор в качестве элементарного вывода белого или черного квадрата ведет после проверки правильности к завершению доказательства утверждения или его части (ветви доказательства).



Р и с. 3. Главное окно программы

F i g. 3. The main window of the program



Модели обучения

3.1 Стратегия обучения

Весь материал для обучения можно разделить на две большие категории. В первую категорию войдут секции, дающие основные теоретические знания и понятия по определению множеств, операций со множествами и алгебре множеств. Основная цель этих секций – развитие и тренировка памяти обучаемого с использованием навыков запоминания. Контроль этих секций, чаще всего, производится с помощью тестирования.

Однако, в программу курса также входит обучение неформальному использованию математического аппарата, и в этом месте возникают дополнительные трудности. К ним относятся технические трудности, связанные с вводом формул и их преобразованием, и трудности, связанные с недостаточным уровнем логического мышления учащегося. Он должен достигнуть цели, разбивая доказательство на части, выбирая методы доказательства для этих частей и конструируя последовательность выводов, ведущих к целевому утверждению. Таким образом, определяется вторая категория секций, связанных с математической методикой проведения доказательства утверждения для множеств.

Выделяются следующие черты, отличающие вторую категорию секций от первой. Во-первых, контроль материала секций второй категории не может быть основан только на тестировании, потому что необходимо проверять умение учащегося использовать различные математические приёмы. Во-вторых, при изучении материала требуется научить студента связывать отдельные приёмы в целенаправленный процесс путём конструирования последовательности изученных приёмов. В-третьих, нужно проконтролировать умение связывать несколько процессов при решении итоговой задачи. Одним из инструментов, используемых в контроле материала секций второй категории, является алгоритм проверки синтаксической правильности ввода и алгоритм контроля правильности вывода.

Материал и контроль взаимодействуют с помощью третьего компонента системы, ответственного за определение объёма материала за один сеанс, и задают набор заданий и их количество, а также другие параметры системы. Именно этот компонент придаёт системе гибкость и отличает АОС от приложения, которое умеет только выдавать текст и набор тестов по нему. Далее следует описание предлагаемого сценария взаимодействия системы и пользователя.

При входе в систему пользователь видит список секций курса, которые делятся на доступные и недоступные в соответствии с планом прохождения курса. Секции должны быть пройдены одна за другой по порядку (принцип линейного обучения). При входе в секцию студенту предоставляется для изучения её материал. После изучения материала он переходит к контролю полученных знаний, для прохождения которого он должен выполнить определённое в секции начальное количество заданий (тестов, упражнений, задач). На каждый вопрос теста предлагается несколько вариантов ответов (обычно 6), случайно выбранных из заранее определённого для каждого задания множества ответов. Среди предлагаемых ответов может быть несколько правильных, или все правильные, или ни одного правильного. Студент должен отметить все правильные ответы. Но если, по его мнению, правильных ответов нет, то он должен выбрать именно такой ответ. Система считает задание

выполненным, если студент отметил все правильные ответы и ни одного неправильного. В противном случае она выводит на экран или один из текстов: «Не все правильные ответы отмечены», «Некоторые из отмеченных ответов не верны», или их комбинацию и предоставляет фрагмент текста материала секции, связанный с совершенной ошибкой. После изучения этого фрагмента или повторного изучения всего материала секции студент имеет возможность повторно ответить на вопрос задания. Если же он опять не выполнит задание, то оно будет заменено на два дополнительных задания. Таким образом, число заданий для прохождения секции может расти. Сеанс работы со студентом будет прекращён при достижении некоторого предельного количества заданий секции, и он сможет вернуться к обучению только после определённого перерыва. Если студент добивается прерывания сеанса несколько раз подряд, то его учётная запись в системе временно блокируется, а сам он вызывается к преподавателю. В случае, когда сначала студент заработал много дополнительных заданий, а затем начал отвечать безошибочно, число контрольных заданий начинает снижаться по некоторой прогрессии. Такой подход заставляет студента внимательно и вдумчиво относиться к материалу секции. Таким образом, организуется не только контроль знаний секции, но и обучение. Только после выполнения всех заданий студент сможет перейти к изучению материала следующей секции.

Однако, существуют секции, материал которых опирается на предыдущие секции и не может быть освоен без безусловного владения материалом этих предыдущих секций. В таких секциях проводится дополнительный контроль знаний предыдущих секций. При этом количество начальных заданий для повторения секции снижается до одного.

Отметим еще, что для настройки АСО определяются настраиваемые параметры обучения. Например, начальное количество контрольных заданий в каждой секции или подсекции, количество добавляемых заданий при ошибочном ответе, количество сокращаемых заданий при каждых 2 последовательных правильных ответах и др.

3.2 Разбиение на секции

Весь материал разобьем на следующие секции и подсекции.

1. *Определение множества.* Тесты на определение верных или неверных ответов.

2. *Отношения множеств:*

2.1. Отношение равенства. Тесты на определение верных или неверных ответов.

2.2. Отношение включения. Тесты на определение верных или неверных ответов.

3. *Операции над множествами:*

3.1 Объединение множеств. Тесты на определение верных или неверных ответов. Упражнения на доказательство и преобразование.

3.2 Пересечение и дополнение множеств. Тесты на определение верных или неверных ответов. Упражнения на доказательство и преобразование.

4. *Алгебра множеств.* Тесты на определение верных или неверных ответов. Упражнения на задание формул множеств.

5. *Методика доказательства утверждений для множеств.*

5.1. Посылки и заключение шага доказательства. Тесты на определение верных или неверных ответов. Упражнения на вывод заключения из заданных посылок. Упражнения на определение посылок для заданного заключения.



5.2. Разбиение на случаи. Тесты на определение верных или неверных ответов относительно ветвей доказательства. Упражнения для разбиения на обычные и альтернативные случаи.

5.3. Прямой метод доказательства. Тесты на определение верных или неверных ответов относительно завершения ветвей доказательства. Упражнения на определение начальной посылки и вывод из нее.

5.4. Косвенный метод доказательства. Тесты на определение верных и неверных ответов относительно ветвей доказательства и их завершения. Упражнения на задание начальной посылки. Упражнения на двойное разбиение на случаи.

6. *Ошибки в доказательстве*. Тесты на определение верных и неверных ответов.

7. *Итоговая секция*. Задача доказательства утверждения для множества.

7.1.. Числовые примеры с общими случаями выполнения всех условий утверждения и с невыполнением каждого из условий.

7.2. Диаграммы Венна с общими случаями выполнения всех условий утверждения и невыполнением каждого из условий.

Заключение

Выразим надежду, что описанный подход к построению автоматизированной обучающей системы доказательству утверждений для множеств будет реализован и покажет эффективность в обучении этому предмету и развитию логико-математического мышления студентов.

Список использованных источников

- [1] *Büchner A.* Moodle 3 Administration. Third Edition. Packt Publishing, 2016. 492 pp.
- [2] *Udaya Sri K., Vamsi Krishna T. V.* E-Learning: Technological Development in Teaching for school kids // International Journal of Computer Science and Information Technologies. 2014. Vol. 5, Issue 5. Pp. 6124-6126. URL: <http://www.ijcsit.com/docs/Volume%205/vol5issue05/ijcsit2014050527.pdf> (дата обращения: 10.08.2019).
- [3] *Kerres M., Annabell Preußler* Mediendidaktik // Enzyklopädie Erziehungswissenschaft Online / Dorothee Meister, Friederike von Gross, Uwe Sander (Hrsg.). Fachgebiet Medienpädagogik. Weinheim/Basel: Beltz Juventa, 2012. DOI: 10.3262/EE018120258
- [4] *Cavus N., Zabadi T.* A Comparison of Open Source Learning Management Systems // Procedia - Social and Behavioral Sciences. 2014. Vol. 143. Pp. 521-526. DOI: 10.1016/j.sbspro.2014.07.430
- [5] *Clark S., Baggaley J.* Technical Evaluation Report. 37. Assistive Software for Disabled Learners // International Review of Research in Open and Distributed Learning. 2004. Vol. 5, No. 3. DOI: 10.19173/irrod.v5i3.198
- [6] *Awang N. B., Darus M. Y. B.* Evaluation of an Open Source Learning Management System: Claroline // Procedia - Social and Behavioral Sciences. 2012. Vol. 67. Pp. 416-426. DOI: 10.1016/j.sbspro.2012.11.346
- [7] *Nagy A.* The Impact of E-Learning // E-Content / P.A. Bruck, Z. Karssen, A. Buchholz, A. Zerfass (Eds.) Springer, Berlin, Heidelberg, 2005. Pp. 79-96. DOI: 10.1007/3-540-26387-X_4
- [8] *Allen I. Elaine, Seaman J.* Changing Course: Ten Years of Tracking Online Education in the United States. Babson Survey Research Group, 2013. Pp. 4. URL: <http://www.onlinelearningsurvey.com/reports/changingcourse.pdf> (дата обращения: 10.08.2019).
- [9] *Burgos D. et al.* Higher Education Learning Methodologies and Technologies Online. First International Workshop, HELMeTo 2019, Novedate CD Italy, June 6-7, 2019, Revised Selected Papers. Springer, Cham, 2019. DOI: 10.1007/978-3-030-31284-8
- [10] *Holzinger A., Kieseberg P., Tjoa A. M., Weipp E.* Machine Learning and Knowledge Extraction. Third IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 International Cross-Domain Conference, CD-MAKE 2019, Canterbury, UK, August 26–29, 2019, Proceedings // LNCS. Vol. 11713. Springer, Cham, 2019. DOI: 10.1007/978-3-030-29726-8
- [11] *Zhang Y., Cristol D.* Handbook of Mobile Teaching and Learning. Springer, Singapore, 2019. DOI: 10.1007/978-981-13-2766-7
- [12] *Kong S., Abelson H.* Computational Thinking Education. Springer, Singapore, 2019. DOI: 10.1007/978-981-13-6528-7
- [13] *Hanna G., Reid D. A., de Villiers M.* Proof Technology in Mathematics Research and Teaching. Springer, Cham, 2019. DOI: 10.1007/978-3-030-28483-1
- [14] *England M., Koepf W., Sadykov T. M., Seiler W. M., Vorozhtsov E. V.* Computer Algebra in Scientific Computing. 21st International Workshop, CASC 2019, Moscow, Russia, August 26-30, 2019, Proceedings // LNCS. Vol. 11661. Springer, Cham, 2019. DOI: 10.1007/978-3-030-26831-2
- [15] *Davenport J. H., Siret Y., Tournier E.* Computer algebra: systems and algorithms for algebraic computation. Academic Press Ltd., GBR. 1988.
- [16] *Gathen J., Gerhard J.* Modern Computer Algebra. Third Edition. Cambridge University Press, 2013. DOI: 10.1017/CBO9781139856065
- [17] *Таранчук В. Б.* Основные функции систем компьютерной алгебры. Минск: БГУ, 2013. 59 с.
- [18] *Ермилова А. В., Рублев В. С.* Проблемы развития математического мышления учащихся на примере обучающей системы по курсу «Алгоритмы и анализ сложности» // Современные информационные технологии и ИТ-образование. 2014. № 10. С. 297-304. URL: <https://www.elibrary.ru/item.asp?id=23020647> (дата обращения: 10.08.2019).
- [19] *Рублев В. С., Юсуфов М. Т.* Автоматизированная система для обучения анализу вычислительной сложности алгоритмов // Современные информационные технологии и ИТ-образование. 2016. Т. 12, № 1. С. 135-145. URL: <https://www.elibrary.ru/item.asp?id=27539228> (дата обращения: 10.08.2019).
- [20] *Рублев В. С., Юсуфов М. Т.* Автоматизированная обучающая система «Анализ вычислительной сложности алгоритмов» (исследование организации 1-ой части проекта) // Современные информационные технологии и ИТ-образование. 2017. Т. 13, № 2. С. 170-178. DOI: 10.25559/SITITO.2017.2.237

Поступила 10.08.2019; принята к публикации 25.10.2019;
опубликована онлайн 23.12.2019.



Об авторах:

Рублев Вадим Сергеевич, профессор кафедры теоретической информатики, Ярославский государственный университет им. П.Г. Демидова (150003, Россия, г. Ярославль, ул. Советская, д. 14), кандидат физико-математических наук, ORCID: <http://orcid.org/0000-0002-0252-9958>, roublev@mail.ru

Вахмянин Даниил Ростиславович, студент кафедры теоретической информатики, Ярославский государственный университет им. П.Г. Демидова (150003, Россия, г. Ярославль, ул. Советская, д. 14), ORCID: <http://orcid.org/0000-0002-8836-0865>, vakhmyanin282@mail.ru

Все авторы прочитали и одобрили окончательный вариант рукописи.

References

- [1] Büchner A. Moodle 3 Administration. Third Edition. Packt Publishing, 2016. (In Eng.)
- [2] Udaya Sri K., Vamsi Krishna T. V. E-Learning: Technological Development in Teaching for school kids. *International Journal of Computer Science and Information Technologies*. 2014; 5(5):6124-6126. Available at: <http://www.ijcsit.com/docs/Volume%205/vol5issue05/ijcsit2014050527.pdf> (accessed 10.08.2019). (In Eng.)
- [3] Kerres M., Annabell Preußler Mediendidaktik. *Enzyklopädie Erziehungswissenschaft Online*. Dorothee Meister, Friederike von Gross, Uwe Sander (Hrsg.). Fachgebiet Medienpädagogik. Weinheim/Basel: Beltz Juventa, 2012. (In Germ.) DOI: 10.3262/EE018120258
- [4] Cavus N., Zabadi T. A Comparison of Open Source Learning Management Systems. *Procedia - Social and Behavioral Sciences*. 2014; 143:521-526. (In Eng.) DOI: 10.1016/j.sbspro.2014.07.430
- [5] Clark S., Baggaley J. Technical Evaluation Report. 37. Assistive Software for Disabled Learners. *International Review of Research in Open and Distributed Learning*. 2004; 5(3). (In Eng.) DOI: 10.19173/irrodl.v5i3.198
- [6] Awang N.B., Darus M.Y.B. Evaluation of an Open Source Learning Management System: Claroline. *Procedia - Social and Behavioral Sciences*. 2012; 67:416-426. (In Eng.) DOI: 10.1016/j.sbspro.2012.11.346
- [7] Nagy A. The Impact of E-Learning. In: Bruck P.A., Karssen Z., Buchholz A., Zerfass A. (Eds.) *E-Content*. Springer, Berlin, Heidelberg, 2005, pp. 79-96. (In Eng.) DOI: 10.1007/3-540-26387-X_4
- [8] Allen I. Elaine, Seaman J. Changing Course: Ten Years of Tracking Online Education in the United States. Babson Survey Research Group, 2013, pp. 4. Available at: <http://www.onlinelearningsurvey.com/reports/changingcourse.pdf> (accessed 10.08.2019). (In Eng.)
- [9] Burgos D. et al. Higher Education Learning Methodologies and Technologies Online. First International Workshop, HELMeTo 2019, Novedate CD Italy, June 6-7, 2019, Revised Selected Papers. Springer, Cham, 2019. (In Eng.) DOI: 10.1007/978-3-030-31284-8
- [10] Holzinger A., Kieseberg P., Tjoa A.M., Weipp E. Machine Learning and Knowledge Extraction. Third IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 International Cross-Domain Conference, CD-MAKE 2019, Canterbury, UK, August 26–29, 2019, Proceedings. *LNCS*, vol. 11713. Springer, Cham, 2019. (In Eng.) DOI: 10.1007/978-3-030-29726-8
- [11] Zhang Y., Cristol D. Handbook of Mobile Teaching and Learning. Springer, Singapore, 2019. (In Eng.) DOI: 10.1007/978-981-13-2766-7
- [12] Kong S., Abelson H. Computational Thinking Education. Springer, Singapore, 2019. (In Eng.) DOI: 10.1007/978-981-13-6528-7
- [13] Hanna G., Reid D.A., de Villiers M. Proof Technology in Mathematics Research and Teaching. Springer, Cham, 2019. (In Eng.) DOI: 10.1007/978-3-030-28483-1
- [14] England M., Koepf W., Sadykov T.M., Seiler W.M., Vorozhtsov E.V. Computer Algebra in Scientific Computing. 21st International Workshop, CASC 2019, Moscow, Russia, August 26-30, 2019, Proceedings. *LNCS*, vol. 11661. Springer, Cham, 2019. (In Eng.) DOI: 10.1007/978-3-030-26831-2
- [15] Davenport J.H., Siret Y., Tournier E. Computer algebra: systems and algorithms for algebraic computation. Academic Press Ltd., GBR. 1988. (In Eng.)
- [16] Gathen J., Gerhard J. Modern Computer Algebra. Third Edition. Cambridge University Press, 2013. (In Eng.) DOI: 10.1017/CBO9781139856065
- [17] Taranchuk V.B. *Osnovnye funktsii sistem komp'yuternoy algebrы* [The main functions of computer algebra systems]. BSU, Minsk, 2013. (In Russ.)
- [18] Ermilova A.V., Rublev V.S. Problemy razvitiya matematicheskogo myshleniya uchashchikhsya na primere obuchayushchey sistemy po kursu «Algoritmy i analiz slozhnosti» [Problems of developing students' mathematical thinking on the example of a training system for the course «Algorithms and complexity analysis»]. *Sovremennye informacionnye tehnologii i IT-obrazovanie = Modern Information Technologies and IT-Education*. 2014; (10):297-304. Available at: <https://www.elibrary.ru/item.asp?id=23020647> (accessed 10.08.2019). (In Russ.)
- [19] Rublev V.S., Yusuf M.T. Automated System for Teaching Computational Complexity of Algorithms Course. *Sovremennye informacionnye tehnologii i IT-obrazovanie = Modern Information Technologies and IT-Education*. 2016; 12(1):135-145. Available at: <https://www.elibrary.ru/item.asp?id=27539228> (accessed 10.08.2019). (In Russ., abstract in Eng.)
- [20] Rublev V.S., Yusuf M.T. Development of the First Part of the Automated System for Teaching Computational Complexity of Algorithms Course. *Sovremennye informacionnye tehnologii i IT-obrazovanie = Modern Information Technologies and IT-Education*. 2017; 13(2):170-178. (In Russ., abstract in Eng.) DOI: 10.25559/SITITO.2017.2.237

Submitted 10.08.2019; revised 25.10.2019; published online 23.12.2019.

About the authors:

Vadim S. Rublev, Professor of the Department of Theoretical Informatics, P.G. Demidov Yaroslavl State University (14 Sovetskaya Str., Yaroslavl 150003, Russia), Ph.D. (Phys.-Math.), ORCID: <http://orcid.org/0000-0002-0252-9958>, roublev@mail.ru

Daniil R. Vahmyanin, Student of the Department of Theoretical Informatics, P.G. Demidov Yaroslavl State University (14 Sovetskaya Str., Yaroslavl 150003, Russia), ORCID: <http://orcid.org/0000-0002-8836-0865>, vakhmyanin282@mail.ru

All authors have read and approved the final manuscript.

