

информационных технологий для гетерогенных технологических платформ информационных инфраструктур с гарантиями качества защиты информации. В методологию включаются математические модели совместного отображения окружающей среды, функциональных и технологических особенностей архитектурной организации распределённых систем, а также аналитические методы определения и анализа динамических характеристик комплексных систем защиты информации.

Список литературы

1. *Птицын А.В.* Генерация системно-аналитического ядра безопасных информационных технологий / А. В. Птицын, Л. К. Птицына. – СПб.: Изд-во Политехн. ун-та. 2011. – 263 с.
2. *Птицын А.В., Птицына Л.К.* Аналитическое моделирование комплексных систем защиты информации. – Гамбург. Saarbrucken: LAP LAMBERT Academic Publishing, 2012. – 293 с.

УДК 371.14

С.А. Рудаков ПРОГРАММЫ НА ЯЗЫКЕ C# В КУРСЕ МАТЕМАТИЧЕСКОЙ ЛОГИКИ

Сергей Аркадьевич Рудаков

rudakov@csu.ru

ФГБОУ ВПО «Челябинский государственный университет», Россия, г. Челябинск

PROGRAMS WRITTEN IN C # IN THE MATHEMATICAL LOGIC COURSE

Sergey Arkadievich Rudakov

Chelyabinsk State University, Russia, Chelyabinsk

Аннотация. Математическая логика входит в программу обучения студентов вузов и колледжей по всем направлениям, связанным с микропроцессорами и программированием. В настоящей работе описывается программа на языке C#, разработанная для курса математической логики. Поставлены задачи для совершенствования программы.

Abstract. Mathematical logic is included in the curriculum of students of high schools and colleges in all fields related to microprocessors and programming. In this paper we describe a program in C #, developed for a mathematical logic course. Targets are set for improving the program.

Ключевые слова: математическая логика; программы; клауза; искусственный интеллект; распараллеливание.

Keywords: mathematical logic; program; clause; artificial intelligence; parallelization.

«Математическая логика – раздел математики, изучающий математические обозначения, формальные системы, доказуемость математических суждений, природу

математического доказательства в целом, вычислимость и прочие аспекты оснований математики» [1].

Математика является наукой с самым высоким уровнем абстракции, что позволяет применять математические модели во всех областях интеллектуальной и материальной деятельности человечества. Исчисление высказываний, появившееся в середине 19 века как составная часть математической логики, стало необходимой компонентой функционирования всех современных микропроцессоров.

Взаимосвязь математической логики и программирования усилилась со времени появления языков программирования высокого уровня. Все инструкции по управлению последовательностью выполнения команд используют выражения математической логики. В 1969 году английский математик Чарльз Хоар предложил формальную систему с набором логических правил, предназначенных для доказательства корректности компьютерных программ. В этой формальной системе, названной логика Хоара, есть аксиомы и правила вывода для всех конструкций простого императивного языка программирования. Таким образом, язык программирования может быть представлен как формальная система логического вывода.

Нерешенные проблемы математики и решенные с помощью длинных трудно проверяемых доказательств побудили математиков к разработке методов автоматического доказательства теорем. Уже на первых этапах компьютеризации человечества появилась программа «Логик-теоретик», реализующая доказательства теорем математической логики. Логик-теоретик был запрограммирован Ньюэллом, Шоу и Саймоном в начале 1956 г. Это была первая эвристическая программа, полностью реализованная на вычислительной машине, первая попытка проникнуть в сложные процессы мышления с помощью исследований в области искусственного разума. Логик-теоретик смогла автоматически доказать 38 законов из книги Б. Рассела и А. Уайтхеда «Принципы математики». В 1960 году они же разработали программу: Общий решатель проблем / General Problem Solver. Она стала более мощным инструментом, чем Логик-теоретик, поскольку могла не только выполнять логические доказательства, но и делать ходы в шахматной игре и строить так называемую «Ханойскую башню». Программа раскладывала исходную задачу на более простые подзадачи, решение которых было возможно достичь [2]. В 1969 году Карлом Хьюитом в Лаборатории Искусственного Интеллекта Массачусетского Технологического Института был разработан функционально-логический язык программирования PLANNER. В 1972 году был создан язык и система логического программирования «Prolog», основанный на языке предикатов математической логики дизъюнктов Хорна. Таким образом, компьютеры оказались замечательным инструментом для реализации исчисления предикатов.

Дисциплины, в которых излагается математическая логика, входят в программу обучения студентов вузов и колледжей по всем направлениям, связанным с микропроцессорами и программированием.

У основной массы студентов затруднения в изучении математической логики вызваны следующими причинами:

- высочайший уровень абстракции;
- строгое соблюдение правил синтаксиса и учет семантики точного языка математической логики;

- отсутствие геометрической интерпретации и наглядных иллюстраций в математической логике.

Программные средства могут помочь в изложении и усвоении математической логики в занимательной, наглядной форме. Например, электронные таблицы могут существенно ускорить построение таблицы истинности для высказывания с большим числом пропозициональных переменных.

Поиск в интернете основных понятий математической логики приводит на страницы форумов с просьбами решения задач по математической логике и дискретной математике. Среди таких задач построение СДНФ и СКНФ по множеству значений булевой функции, построение таблиц истинности для заданных высказываний, минимизация представления булевых функций, доказательства истинности клауз различными методами (например, резолюции, натурального исчисления). Существуют программы, которые в режиме «онлайн» решают эти задачи, например [3].

У преподавателя возникает потребность в формировании большого количества разнообразных задач для контрольных и тестов, а также их решений в виде доступном для понимания студентами при самостоятельной работе. Существующие программы не предназначены для преподавателя.

Для разработки программ, удовлетворяющих потребностям преподавателя, был выбран язык программирования C# на платформе .NET Framework версии 4.5.50938, реализованный в Microsoft Visual Studio Ultimate 2012 версии 11.0.50727.1.

Указанная версия реализации языка C# обладает большим количеством методов работы со строками и коллекциями, а также возможностью разработки многопоточных приложений, позволяющих ускорить выполнение программ на многоядерных процессорах.

Цели разработки программы:

- а) помочь студентам в усвоении материала по дискретной математике и математической логике;
- б) помочь преподавателю в изложении материала в доступной и наглядной форме;
- в) решение логических задач, проведение доказательств различными методами математической логики.

На настоящем этапе разработки окно программы выглядит так, как показано на рис. 1.

В текстовую строку можно вводить высказывания с использованием букв латинского алфавита и общепринятых обозначений логических операций. Высказывание можно проверить на истиннозначность с помощью таблиц истинности. Для синтаксического анализа использована обратная польская запись.

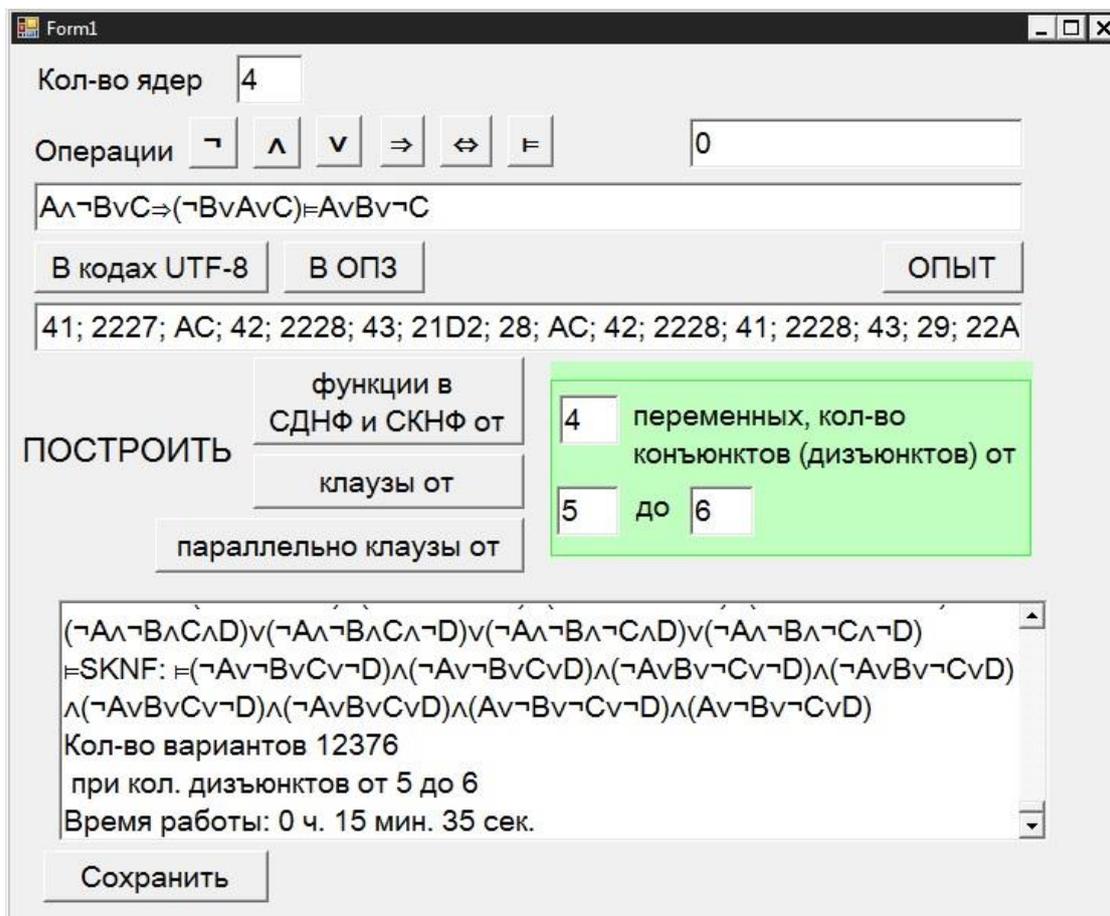


Рис. 1. Скрин-шот окна программы

С помощью перебора значений булевых функций программа создает огромное количество вариантов корректных клауз и высказываний для использования в тестах и контрольных. Программа может сохранять построенные варианты в форматах txt, doc, html. Количество вариантов зависит от числа переменных и растет со скоростью $2^{(2^n)}$, где n – количество пропозициональных переменных (A, B, C,...). Следующая таблица отражает этот стремительный рост минимального количества возможных вариантов в зависимости от числа переменных.

Таблица 1. Зависимость количества формул от числа пропозициональных переменных

Кол-во переменных	3	4	5	6	7
Кол-во вариантов	256	65536	4294967296	1,84467E+19	3,40282E+38

Перебор всевозможных формул уже при 10 пропозициональных переменных не под силу суперкомпьютерам. Поэтому необходим какой-то фильтр для выбора подходящих вариантов. Пока отбор идет только по числу дизъюнктов в СДНФ. Следующая таблица отражает рост времени расчета при 4 переменных в зависимости от числа дизъюнктов в формуле.

Таблица 2. Зависимость времени расчета от числа дизъюнктов

Кол-во вариантов	680	380	6188	12376	19448	24310
Кол-во дизъюнктов	2-3	3-4	4-5	5-6	6-7	7-8
Время расчета	1 с.	15 с.	2 м. 57 с.	8 м. 57 с.	15 м. 31 с.	24 м. 29с.

Параллельные вычисления с использованием процессора с 4 ядрами позволяют ускорить вычисления в среднем в 2,3 раза.

Столь быстрое увеличение времени расчета истинных клауз иллюстрирует трудности, возникающие перед искусственным интеллектом при прямом переборе истинных клауз при решении реальных логических задач с большим числом пропозициональных переменных.

Для завершения начального этапа разработки программы следует решить следующие задачи:

- Разработка справочной системы по математической логике.
- Реализация логических законов для произвольных форм высказываний (включая минимизацию высказывания).
- Реализация различных методов доказательства клауз.
- Переход к логике предикатов.

Работа над программой не может иметь завершения, как и работа над созданием искусственного интеллекта.

Список литературы

1. Математическая логика – Википедия [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BB%D0%BE%D0%B3%D0%B8%D0%BA%D0%B0.

2. Аллен Ньюэлл (Ньюелл): эвристическое программирование, моделирование [Электронный ресурс]. – Режим доступа: <http://vikent.ru/author/298>.

3. Онлайн инструменты по математической логике [Электронный ресурс]. – Режим доступа: <http://tablica-istinnosti.ru/ru>.

УДК 371.14

Т.Н. Рудакова

**ИНФОРМАЦИОННО-КОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ В
ИНДИВИДУАЛЬНОМ ОБУЧЕНИИ МАТЕМАТИКЕ**

Рудакова Татьяна Николаевна

rutani@list.ru

*ФГБОУ ВПО «Южно-Уральский государственный университет» НИУ, Россия,
г. Челябинск*

**INFORMATION AND COMMUNICATION TECHNOLOGY
IN INDIVIDUAL MATHEMATICS TEACHING**

Rudakova Tatiana Nikolaevna

South Ural State University, Russia, Chelyabinsk

Аннотация. Исследуется применение информационно-коммуникационных технологий в индивидуальном обучении. Проведенный анализ показывает, что система репетиторства является необходимой формой обучения, направленной на повышение качества образования учащихся школ и студентов.