БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра интеллектуальных информационных технологий

**Отчёт**

по лабораторной работе №2

по дисциплине

«Логические основы интеллектуальных систем»

Выполнил студент группы 021702

Локтев К.А.

Проверил

Ивашенко В.П.

Минск 2023

Вариант 6

**Тема:** преобразования и интерпретация формул языка логики высказываний.

**Цель:** приобрести навыки программирования алгоритмов интерпретации и преобразований формул языка логики высказываний.

**Задача:** написать программу, способную определять, являются ли две переданные ей строки равносильными формулами сокращённого языка логики высказываний.

**Входные данные:** две строки, содержащие тестируемые формулы

**Выходные данные:** логическая истина, если переданные строки являются равносильными формулами сокращённого языка логики высказываний; логическая ложь, если переданные строки таковыми не являются, либо если одна или обе строки не являются формулами сокращённого языка логики высказываний.

**Реализация**

Для реализации был выбран язык JavaScript версии ES2022.

Для реализации были использованы методы парсера формул сокращённого языка логики высказываний, полученного в результате выполнения первой лабораторной работы по дисциплине.

**Использованные структуры данных:** массив, ассоциативный массив (на основе массива и хэш-функции).

**Неформальное описание основного алгоритма программы:**

1. В метод areFormulasEquivalent() передаются входные строки
2. Если хотя бы одна из переданных строк не является формулой сокращённого языка логики высказываний, алгоритм завершает работу со значением «логическая ложь»
3. **combinedAtomicFormulae** = <массив всех уникальных атомарных формул для переданных строк, упорядоченный по алфавиту>
4. **currInputDecimal** = 0, **inputsQuantity** =
5. **currInput** = <**currInputDecimal** в двоичной системе счисления, интерпретированный, как массив нулей и единиц>
6. **argValuesObj** = <ассоциативный массив значений атомарных формул, составленный на основе **currInput** и **combinedAtomicFormula**, где ключ – это атомарная формула, а значение, лежащее по ключу – это значение соответствующей атомарной формулы> (например, если **combinedAtomicFormulae** = [“A”, “C”, “F”] и **currInput** = [1, 1, 0], то **argValuesObj** = { “A”: 1, “C”: 1, “F”: 0 })
7. Вычисляются значения обеих переданных формул при значениях атомарных формул, зафиксированных в **argValuesObj**
8. Если вычисленные значения отличаются, алгоритм завершает работу с выходным значением «логическая ложь»
9. Если **currInputDecimal** == **inputsQuantity** - 1, алгоритм завершает работу с выходным значением «логическая истина»
10. **currInputDecimal** = **currInputDecimal** + 1
11. Алгоритм переходит к пункту 5

**Блок-схемы алгоритмов:**

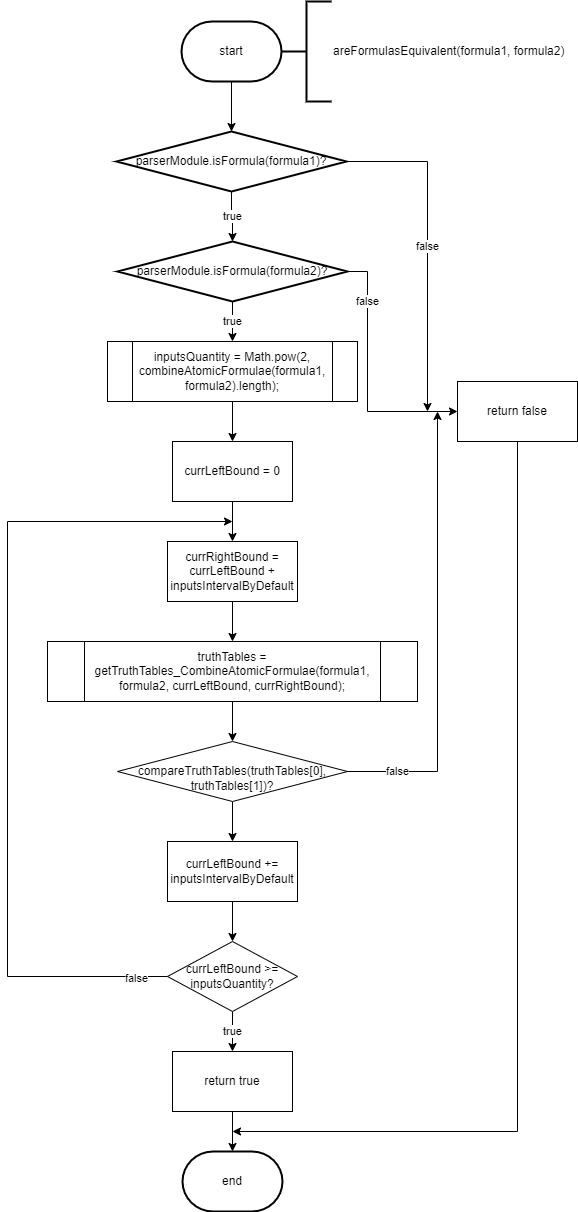
****

Рисунок 1. Блок-схема алгоритма проверки равносильности двух формул

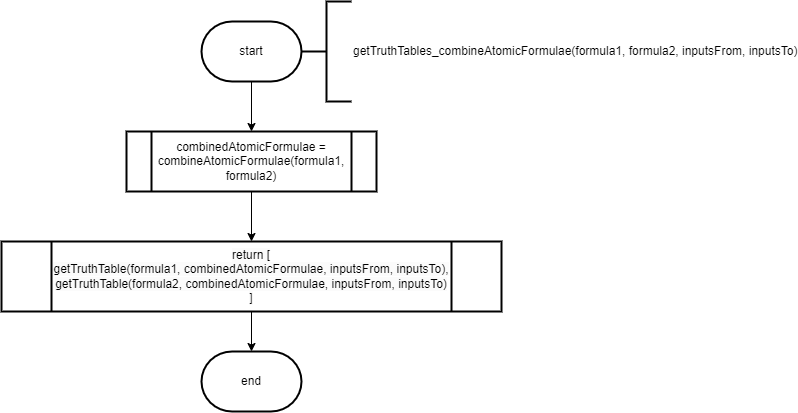
****

Рисунок 2. Блок-схема алгоритма получения совмещённых таблиц истинности

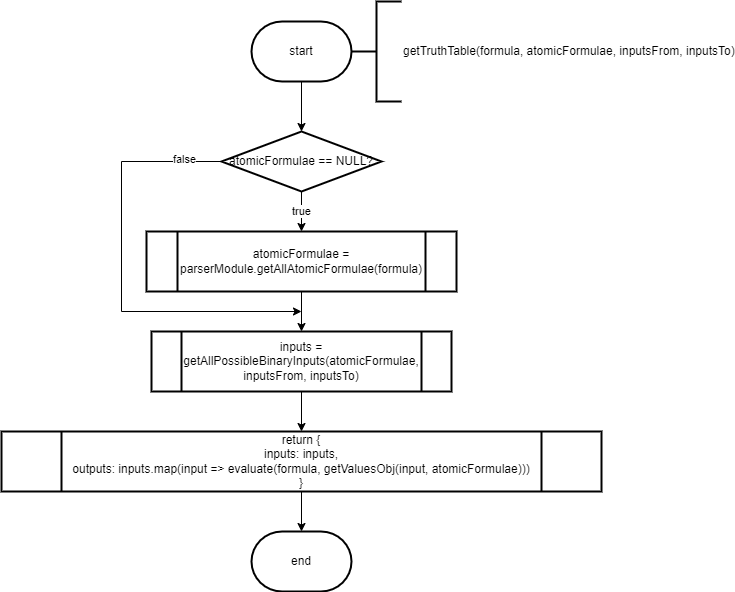
****

Рисунок 3. Блок-схема алгоритма получения таблицы истинности

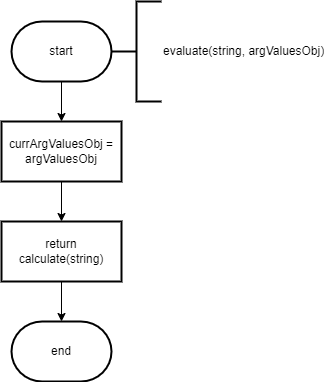
****

Рисунок 4. Блок-схема алгоритма получения значения формулы с заданными значениями атомарных формул

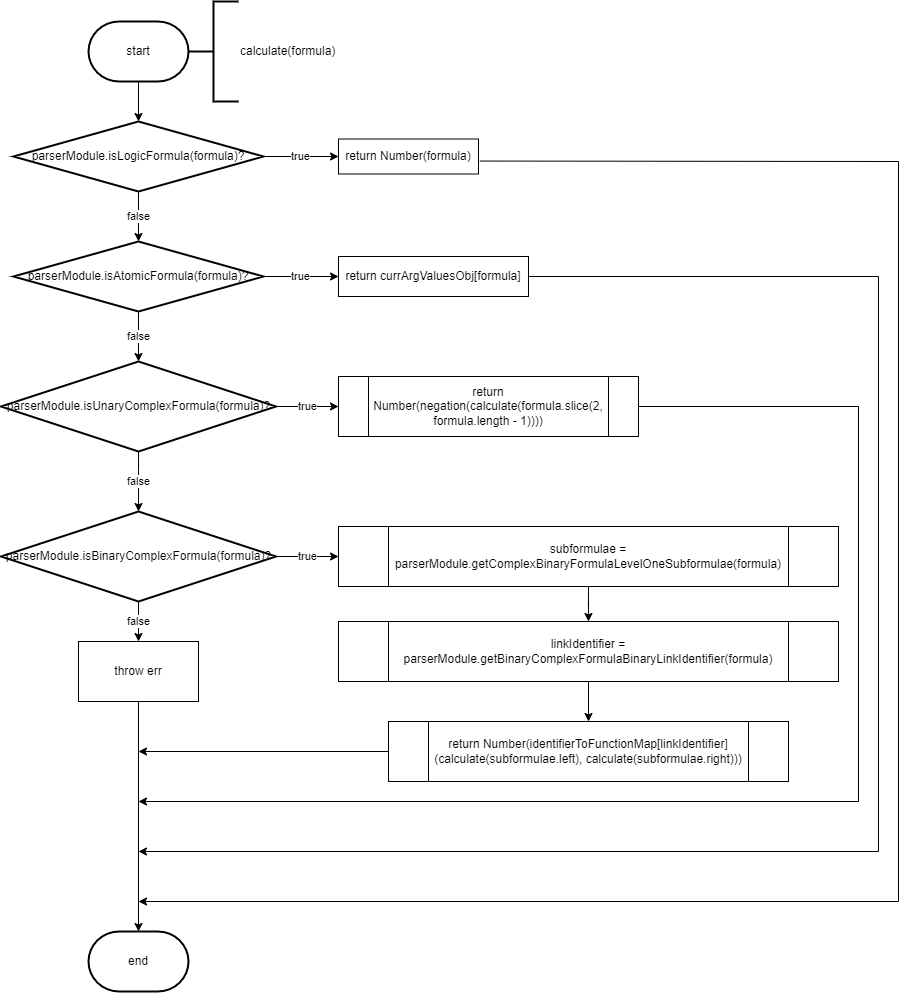
****

Рисунок 5. Блок-схема алгоритма вычисления значения формулы

**Примеры результатов работы программы:**

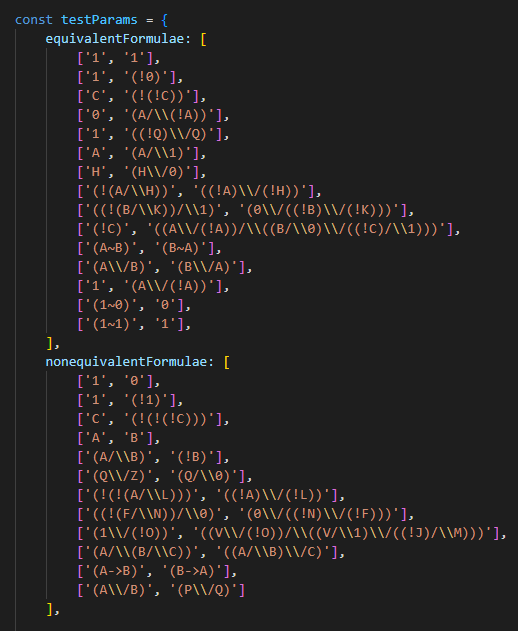


Рисунок 6. Тестовые пары формул

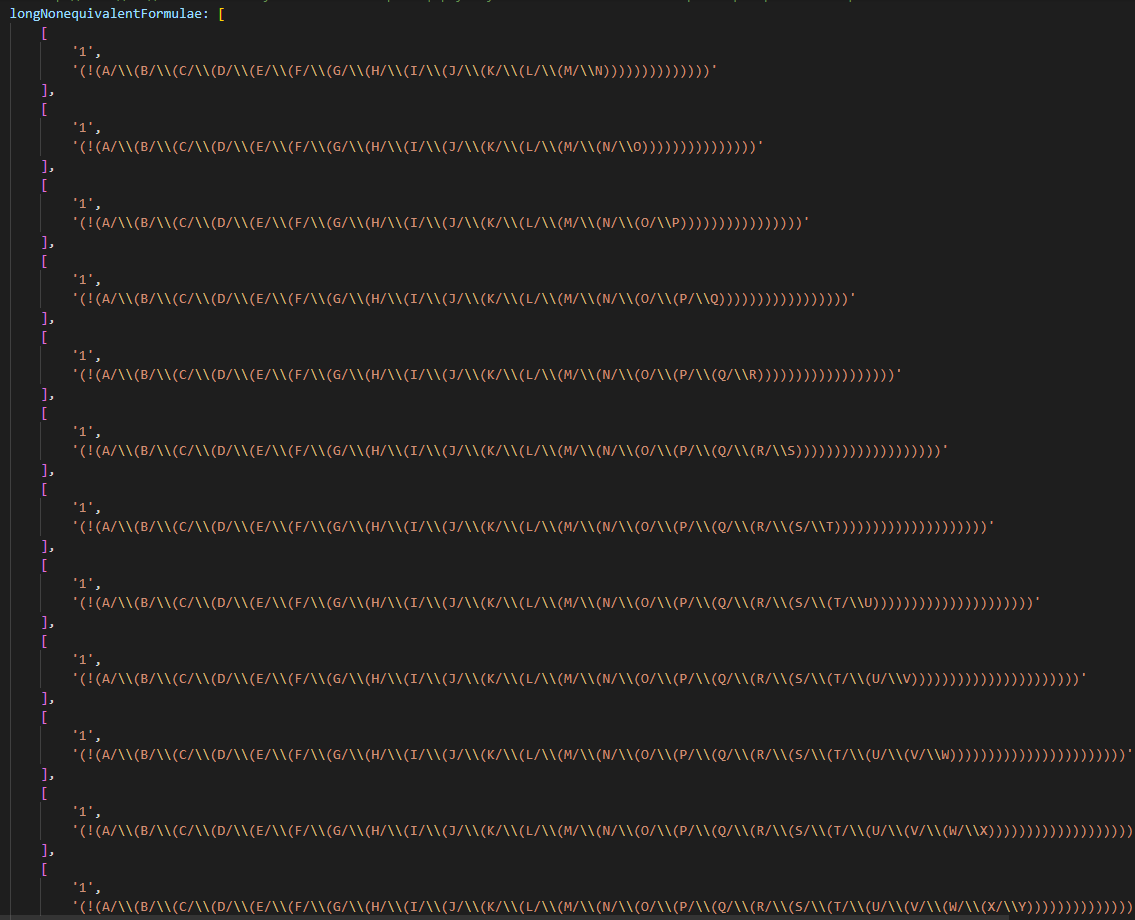


Рисунок 7. Тестовые пары формул с большим общим количеством уникальных атомарных формул

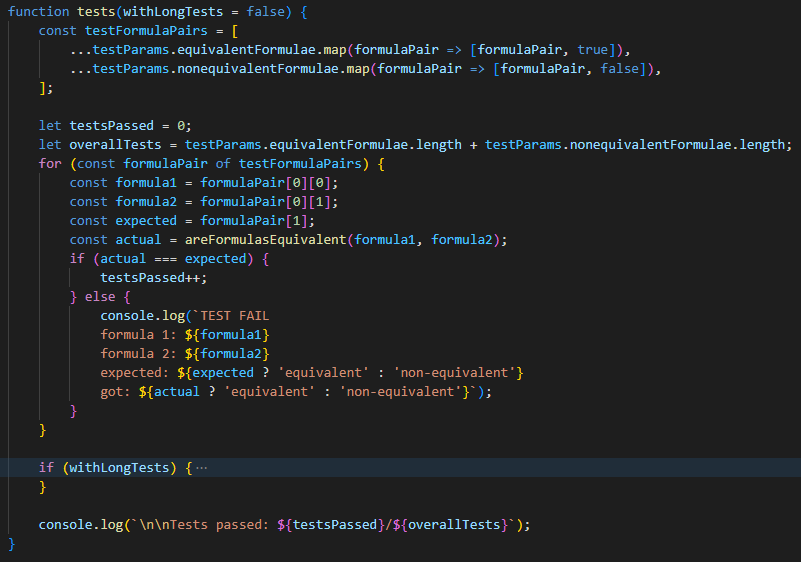


Рисунок 8. Функция для тестирования написанной программы

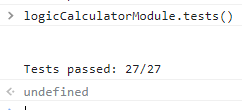


Рисунок 9. Результат выполнения функции tests (при withLongTests = false)

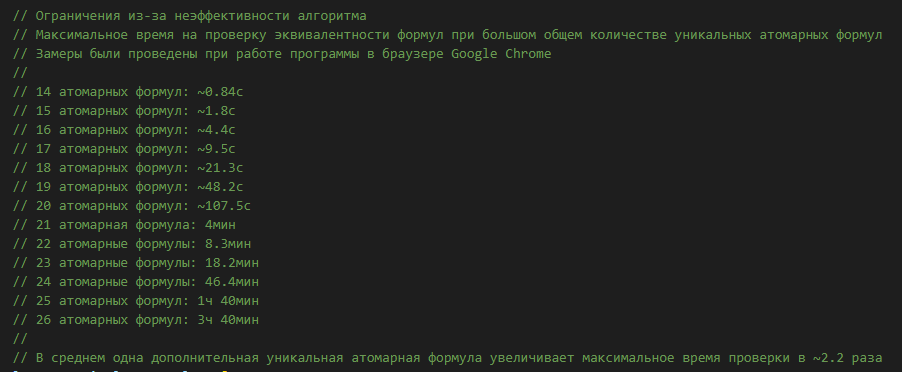


Рисунок 10. Примечание относительно скорости работы алгоритма при большом общем количестве атомарных формул

**Вывод:** в ходе лабораторной работы на основе программы, разработанной в рамках предыдущей лабораторной работы, была разработана программа, способная определять, являются ли две переданные ей формулы сокращённого языка логики высказываний равносильными. Приобретены навыки программирования алгоритмов интерпретации и преобразований формул языка логики высказываний.