Учреждение образования

“БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ”

Кафедра интеллектуальных информационных технологий

**Отчёт по лабораторной работе №1**

**по курсу «Логические основы интеллектуальных систем»**

**«Решение логических задач на языке логики высказываний»**

Выполнил студент группы 721701: Кузьмич Ю.С.

Проверил: Бобков А.В.

**МИНСК 2020**

**Описание лабораторной работы**

*Вариант* ***А****.*

Целью лабораторной работы является реализация программы, которая подсчитывает количество подформул в формуле логики высказываний.

**Постановка задачи**

На входе имеется строка. Необходимо подсчитать количество подформул в формуле логики высказываний.

Язык реализации – JavaScript.

**Формулой** называется правильный текст на языке логики высказываний. **Подформулой** называется подстрока формулы, являющаяся формулой. Формула считается подстрокой/подформулой самой себя.

**Грамматика языка логики высказываний.**

<константа>::=1|0

<символ>::=A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z

<отрицание>::=!

<конъюнкция>::=&

<дизъюнкция>::=|

<импликация>::=->

<эквиваленция>::=~

<открывающая скобка>::=(

<закрывающая скобка>::=)

<бинарная связка> ::= <конъюнкция>|<дизъюнкция>|<импликация>|<эквиваленция>

<атом>::= <символ>

<унарная сложная формула>::= <открывающая скобка><отрицание><формула><закрывающая скобка>

<бинарная сложная формула>::= <открывающая скобка><формула><бинарная связка><формула><закрывающая скобка>

<формула>::= <константа>|<атом>|<унарная сложная формула>|<бинарная сложная формула>

**Схема алгоритма**

1. Функция *run()*

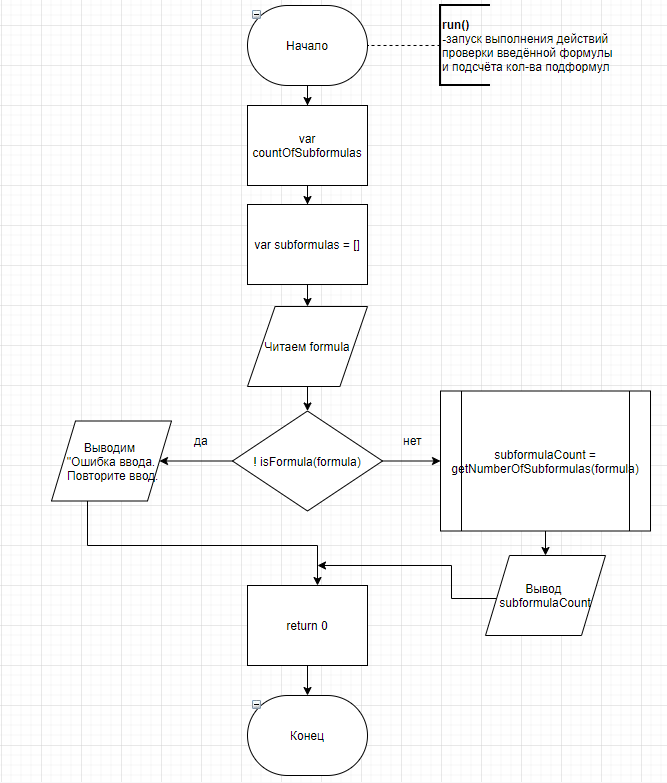


Рисунок 1 – функция run()

1. Функция *isFormula(formula)*

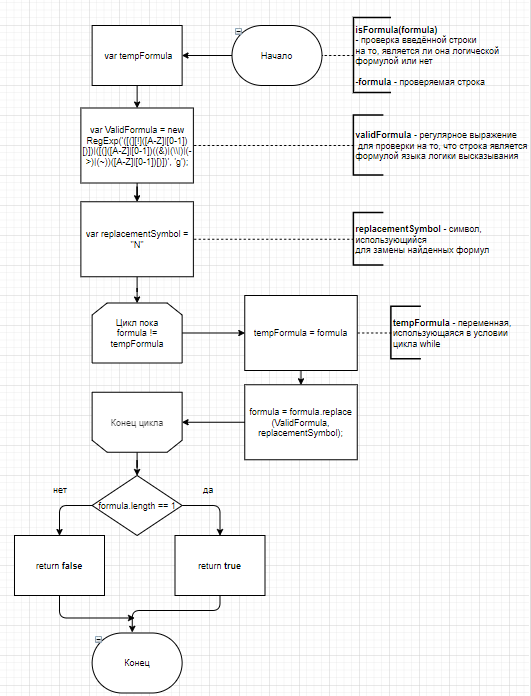


Рисунок 2 – функция isFormula(formula)

1. Функция *getNumberOfSubformulas(formula)*

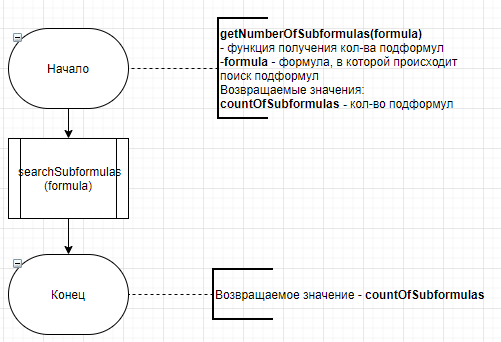


Рисунок 3 – функция getNumberOfSubformulas(formula)

1. Функция *searchSubformulas(formula)*

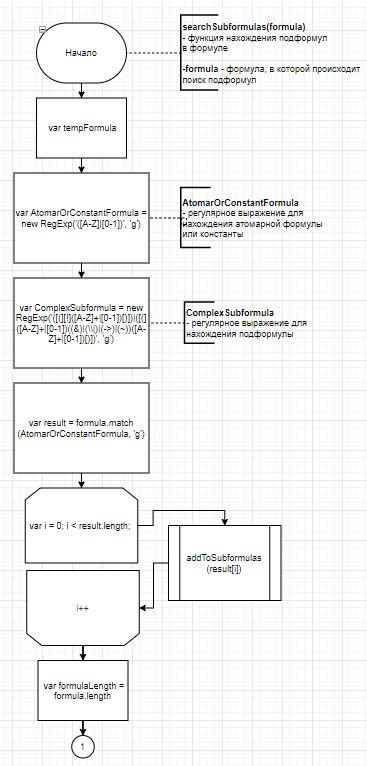


Рисунок 4 – функция searchSubformulas(formula) (начало)

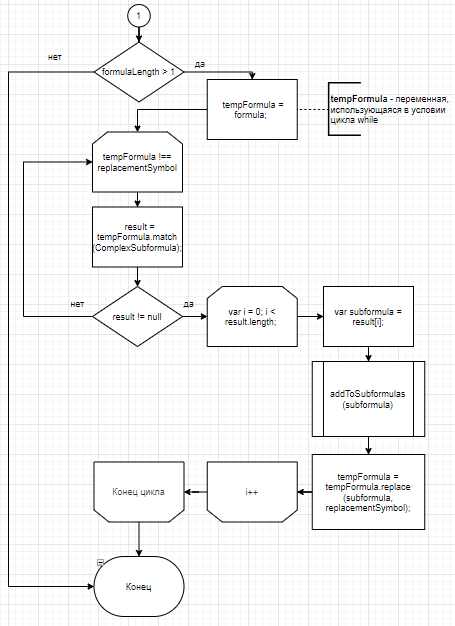


Рисунок 5 – функция searchSubformulas(formula) (конец)

1. Функция *addToSubformulas(subformula)*

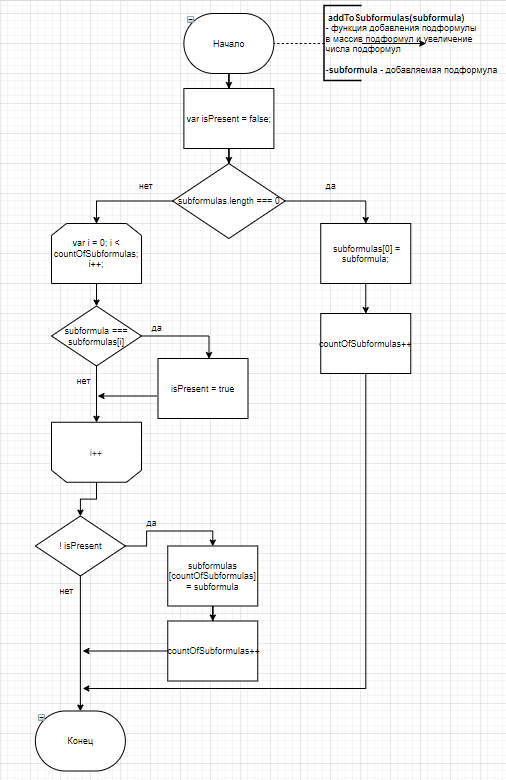


Рисунок 6 – функция addToSubformulas(subformula)

**Тесты:**

1. Тест 1:

*Вход: ((!A)->(!B))*

*Выход:* ***5***

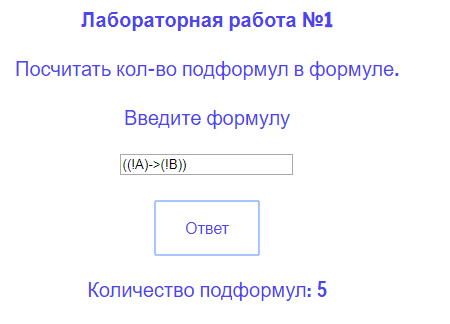


Рисунок 6 – Тест 1

1. Тест 2:

*Вход: ((A&B)|(!C))*

*Выход:* ***6***

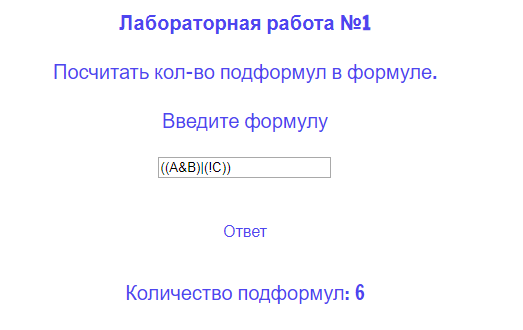


Рисунок 7 – Тест 2

1. Тест 3:

*Вход: ((A~B)|(C->(X|Z)))*

*Выход:* ***9***

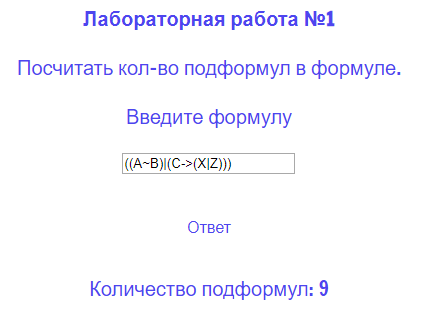


Рисунок 8 – Тест 3

1. Тест 4:

*Вход: ((A&B)&(!C))*

*Выход:* ***6***

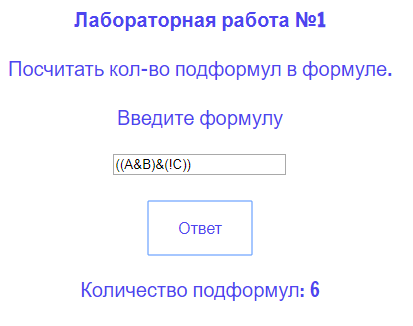


Рисунок 9 – Тест 4

1. Тест 5:

*Вход: (C|((!A)&C))*

*Выход:* ***5***

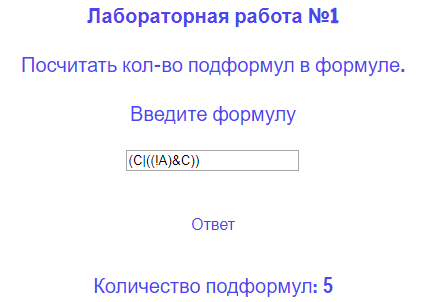


Рисунок 10 – Тест 5

**Вывод:**

В рамках лабораторной работы была реализована программа подсчёта количества подформул в формуле логики высказываний. Также было проведено тестирование программы и составлены схемы алгоритмов в виде блок-схемы.