



Лабораторная работа 1

- (варианты 0, 1, 2, 3, 4) Реализовать алгоритм развертки на n шагов терма по правилам переписывания, определённым TRS. Развертка может быть неоднозначной, поэтому результат — список всех возможных результатов переписывания.
- (варианты 5, 6, 7, 8, 9) Реализовать алгоритм проверки неразрешимости соответствия Поста при анализе образов строк в линейной целочисленной арифметике.



Лабораторная работа 2

- (варианты 0, 1, 2, 3, 4) Дано регулярное выражение. Определить, является ли синтаксический моноид его языка аperiodическим с периодом меньше k , и содержит ли он циклы размерности меньше k . Входные данные: регулярное выражение и значение k .
- (варианты 5, 6, 7, 8, 9) По конечному автомату методом исключения состояний построить регулярное выражение так, чтобы порядок исключения состояний был оптимальный (т.е. алфавитная длина итоговой регулярки была минимальной среди всех, получающихся из данного автомата методом исключения состояний в ином порядке). Конечный автомат описывается набором: стартовое состояние, список финальных состояний через запятую, список правил каждое с новой строки. Пример описания:

Q

Q1, Q3

(Q, a) \rightarrow Q1

(Q1, a) \rightarrow QQ

(Q1, b) \rightarrow Q3

(Q3, b) \rightarrow Q

(QQ, b) \rightarrow Q1



Лабораторная работа 4

- (варианты 0, 1, 2, 3, 4) По грамматике G и регулярке R построить описание слов в регулярном языке, принадлежащих языку грамматики.
- (варианты 5, 6, 7, 8, 9) По SLR(1)-грамматике построить PDA для языка её дополнения.



ЛР 1-1

- TRS задаётся как в лабораторной номер 1 варианты 2, 4, 5, но без ограничения на местность конструкторов.
- Значение n и входное слово для переписывания — дополнительные параметры.
- Исходный терм может переписываться с любой позиции, например, $g(x, x) \rightarrow h(h(x))$ может превращать за один шаг переписывания $g(g(t, t), g(t, t))$ как в $h(h(g(t, t)))$, так и в $g(h(h(t)), g(t, t))$. $g(g(t, t), h(h(t)))$.



ЛР 1-2

- ПСП (списки домино) — список (Φ_1, Φ_2) , перечисленных каждая с новой строки. Например

(ab, a)

(aa, b)

(bb, a)

Считаем, что нет домино с пустыми позициями (т.е. в каждой паре оба слова содержат хотя бы одну букву).

- Если ПСП имеет решение W , то в этом решении определены константы M_i — количество вхождений i -го домино, а также $N_{i,j}$ — количество соседств домино типа i от домино типа j слева. Их значения определяются, в том числе, уравнениями на число вхождений букв в слово W , а также пар букв в слово W . Например, для пары домино выше в каждом решении должны выполняться равенства:

$$(a) \quad 2 \cdot M_2 = M_3$$

$$(b) \quad M_1 + 2 \cdot M_3 = M_2$$

В целых положительных значениях M_i решений этой системы не существует, поэтому ПСП решения не имеет.



ЛР 1-2

- ПСП (списки домино) — список (Φ_1, Φ_2) , перечисленных каждая с новой строки.
Считаем, что нет домино с пустыми позициями (т.е. в каждой паре оба слова содержат хотя бы одну букву).
- Если ПСП имеет решение W , то в этом решении определены константы M_i — количество вхождений i -го домино, а также $N_{i,j}$ — количество соседств домино типа i от домино типа j слева. Их значения определяются, в том числе, уравнениями на число вхождений букв в слово W , а также пар букв в слово W .
- Используя SMT-решатель, проверить, существуют ли подходящие значения M_i и $N_{i,j}$, построив SMT-модель, описывающую уравнения на них в smtlib-языке (для $N_{i,j}$ ещё нужно определить уравнения, связывающие их число с M_i и M_j , и учесть, что крайние домино соседствуют только с одним).