

Отчет по лабораторной работе № 6 по курсу “Фундаментальная информатика”

Студент группы: М8О-106Б-21 Петров Илья Олегович, № по списку: 16

Контакты e-mail, telegram, skype: gtgr3000@gmail.com

Работа выполнена: «13» ноября 2021 г.

Преподаватель: каф. 806 Дубинин Алексей Владимирович

Отчет сдан « » _____ 20__ г., итоговая оценка _____

Подпись преподавателя _____

1. Тема:

Конструирование диаграмм Тьюринга

2. Цель работы:

Разработать диаграмму Тьюринга решения задачи в среде интерпретатора *JDT* или *VisualTuring 2.0* с использованием стандартных машин (x , 1 , R , L , K_n , a_i) и вспомогательных машин, определяемых поставленной задачей.

3. Задание (вариант № 17):

Условие задачи: Вычисление предиката взаимной простоты двух чисел в натуральной системе счисления.

Формат входных данных: Во входной строке через пробел записаны два целых беззнаковых числа произвольной длины в натуральной системе счисления

Формат результата: Ответ (1 -- если исходные числа взаимно простые, 0 -- если нет) должен выводиться через пробел после входных данных, причём модификация исходных данных запрещена.

4. Оборудование (студента): -----

5. Программное обеспечение (лабораторное): -----

6. Идея, метод, алгоритм решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)

Самым простым способом вычисления предиката взаимной простоты двух чисел является применение Алгоритма Евклида, который сводит решение задачи к проверке истинности выражения “НОК(a , b) = Γ ”.

НОК(a , b) – это рекурсивная функция, которая прекращает своё выполнение, когда после очередного выполнения выполняется условие “ $a_i \% b_i = 0$ ”, затем возвращается a_i , что и является результатом НОК(a , b).

НОК основывается на вычислении остатка от деления числа a на b . Для этого нужно написать подмашину, которая будет вычислять остаток.

Подмашина будет основываться на одном простом факте, что в унарной системе счисления остатком от деления левого числа на правое будет являться число, получающееся путём вычитания из первого числа второе, пока выполняется условие $b \leq a$.

Если остаток от деления равен 0, то результатом выполнения подмашины будет являться '0'.

Поскольку функция НОК рекурсивна, для её выполнения нужно копировать аргументы, чтобы не затереть входные данные.

Поскольку между входными данными и аргументами/результатами функции нет никакой разницы, то было решено использовать '0' в качестве разделителя (аналогично '#' [диез]).

Доп. символы использованы не были ($A = \{\lambda, I, 0, 1\}$).

7. Сценарий выполнения работы [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

План работы:

- Изучить свойства взаимно простых чисел;
- Выбрать алгоритм нахождения ответа ($\text{НОД}(a, b) = I$);
- Выбрать алгоритм нахождения НОД (Алгоритм Евклида);
- Составить подмашину, которая находит остаток от деления (Modulo);
- Составить машину, которая найдёт ответ, используя Алгоритм Евклида и подмашину Modulo;

Тесты:

Входные данные	Выходные данные	Описание тестируемого случая
I I	I I 1	Оба числа являются простыми
I II	I II 1	Одно число имеет нечётное число знаков, другое -- чётное
II I	II I 1	Порядок входных данных не влияет на ответ
II II	II II 0	Два чётных числа
IIIIII III	IIIIII III 0	Одно число имеет чётное число знаков, другое -- нечётное
III III	III III 0	Два нечётных числа
IIIIIIIIII IIIII	IIIIIIIIII IIIII 1	Два числа являются простыми, числа имеют разную длину

8. Распечатка протокола (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

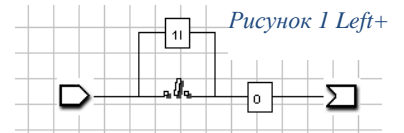
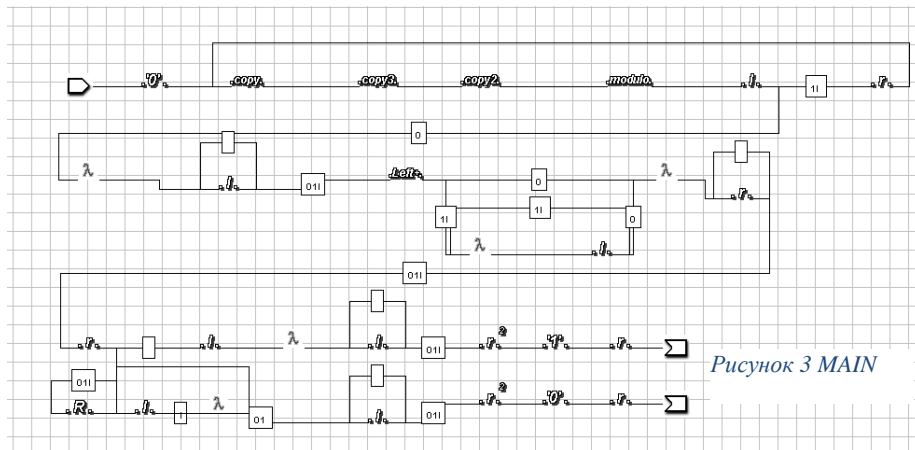


Рисунок 2 Right+

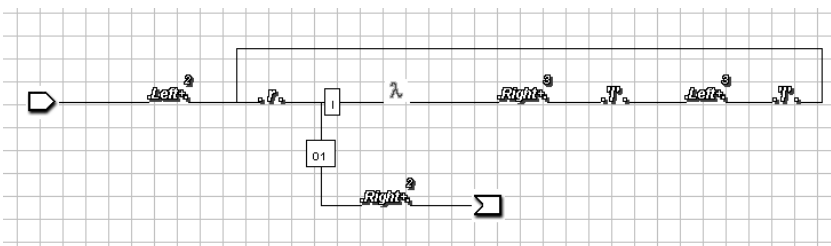
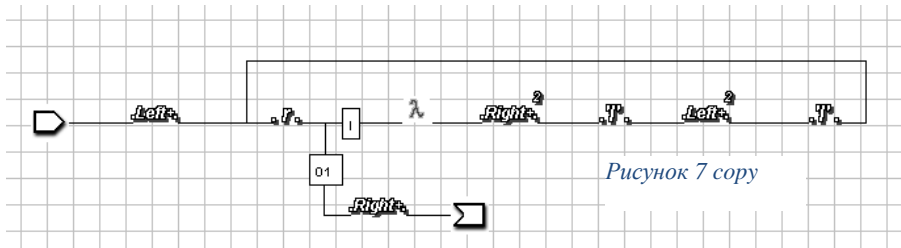
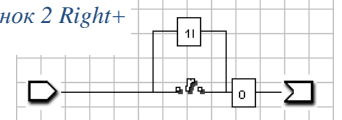


Рисунок 4 сору2

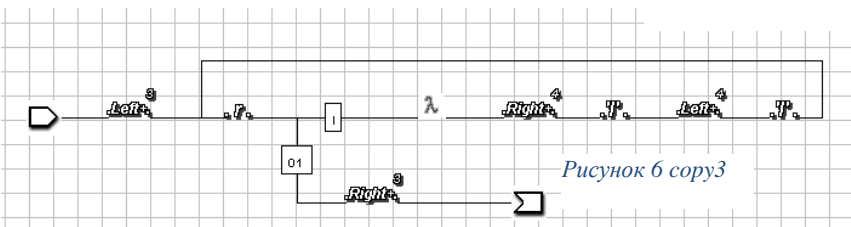


Рисунок 6 сору3

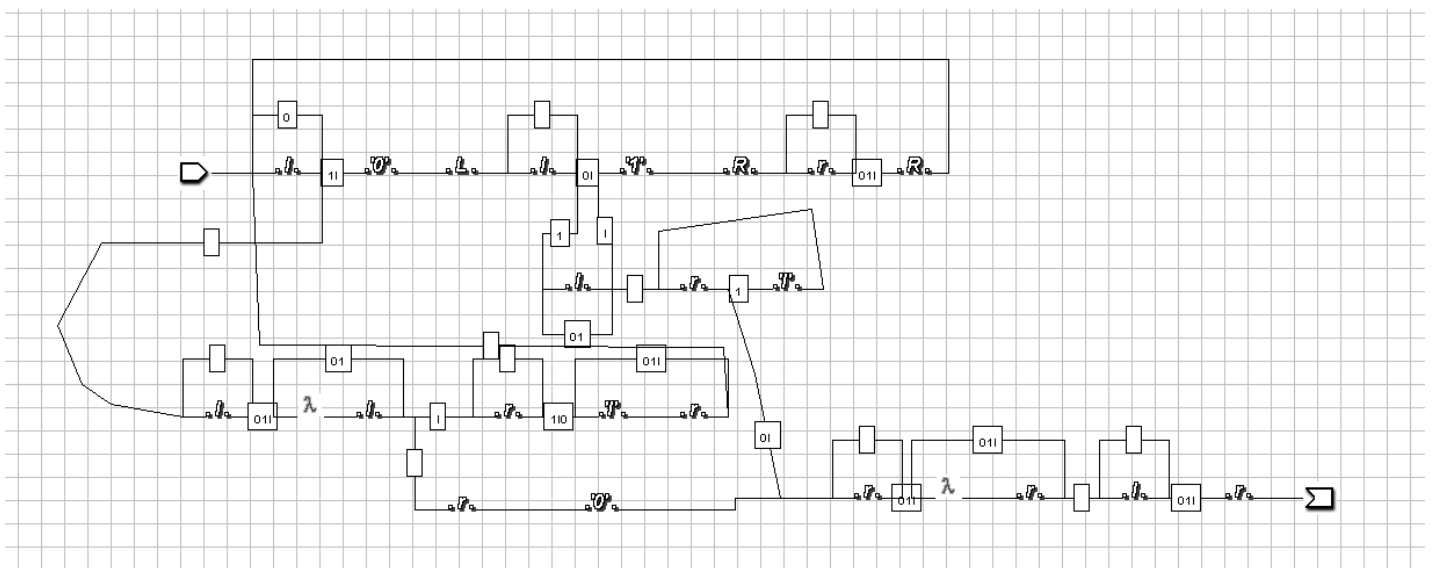


Рисунок 5 modulo

9. Дневник отладки должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание
1	дом	13.11. 2021	11:00	Все тесты пройдены	Не требуется	

10. Замечания автора по существу работы: -----

11. Выводы

Я научился составлять Диаграммы Тьюринга, оценил простоту проектирования алгоритмов в визуальной (наглядной) среде. В отличие от текстового принципа разработки алгоритма, диаграммы имеют большое преимущество – с ростом сложности алгоритма, его читабельность остаётся прежней.

В реальном мире, по моему мнению, диаграммы лучше всего решают проблемы на этапе планирования/прототипирования большой задачи, когда ещё нет представления реального кода, но при этом уже есть идея.

Конструирование диаграмм дало мне опыт в визуализации алгоритмов для лучшего понимания ходов достижения результата.

Недочёты при выполнении задания могут быть устранены следующим образом:

Подпись студента

