|  |  |
| --- | --- |
|  | **Отчёт по лабораторной работе** № 5  по курсу: Фундаментальная информатика  студента группы: M8O-105Б-21 Козлова Егора Сергеевича , № по списку: 10  Контакты: e-mail iamaghoulzxc@gmail.com  Работа выполнена: “17“ октября 2021 г.  Преподаватель: каф. 806 В. К. Титов  Входной контроль знаний с оценкой  Отчёт сдан: “2“ ноября 2021 г., итоговая оценка  Подпись преподавателя |

1. **Тема**: программирование машин Тьюринга
2. **Цель работы**: освоение составления программ МТ в четверках и пятерках. Выполнение программ МТ в среде turun
3. **Задание** (*вариант №* 28):

Перевод числа из двоичной СС в натуральную СС.

*Пример 1010 → 1111111111.*

1. **Оборудование** (*лабораторное*):

*Не использовалось*

*Оборудование ПЭВМ студента, если использовалось:*

Процессор AMD Ryzen 7 5800H @ 8x 3.2GHz , ОП 15429 МБ, НМД 1024 ГБ. Монитор: встроенный (1920x1080)

1. *Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства Linux, наименование: Ubuntu версия 20.04.3 LTS x86\_64 \_\_

Интерпретатор команд: bash версия 5.0.17

Редактор текстов: Emacs версия 26.3

Утилиты операционной системы:

Прикладные системы и программы: turun

Местонахождения и имена файлов программ и данных: /bin

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

//алгоритм копирования (вспомогательный алгоритм, чтобы не изменять входные данные)

q0 λ->\* r q1 //установка разделителя

q1 \*->\* l q1 //установка головки на последнем символе скопированного слова

q1 \*->\* l q1 //установка головки на последнем символе скопированного слова

q1 λ->\* l q2 //установка разделителя

q2 \*->\* l q2 //движение в начало слова

q2 1->1 l q2 //движение в начало слова

q2 0->0 l q2 //движение в начало слова

q2 λ->λ r q3 //смена состояния

q3 1->λ r q5 //стираем единицу и запоминаем ее с помощью состояния

q3 0->λ r q7 //стираем ноль и запоминаем его с помощью состояния

q3 \*->λ r q11 //удаление разделителя и переход к основному алгоритму

q5 1->1 r q5 //установка головки после разделителя и ответа

q5 0->0 r q5 //установка головки после разделителя и ответа

q5 \*->\* r q5 //установка головки после разделителя и ответа

q5 λ->1 l q6 //ставим скопированный символ + возврат головки к пробелу + запоминаем

q6 1->1 l q6 //возврат головки к пробелу

q6 0->0 l q6 //возврат головки к пробелу

q6 \*->\* l q6 //возврат головки к пробелу

q6 λ->1 r q3 //возвращаем единицу на место

q7 1->1 r q7 //установка головки после разделителя и ответа

q7 0->0 r q7 //установка головки после разделителя и ответа

q7 \*->\* r q7 //установка головки после разделителя и ответа

q7 λ->0 l q9 //ставим скопированный символ + вовзрат головки к пробелу + запоминаем

q9 1->1 l q9 //возврат головки к пробелу

q9 0->0 l q9 //возврат головки к пробелу

q9 \*->\* l q9 //возврат головки к пробелу

q9 λ->0 r q3 //возвращаем ноль на место

q11 \*->\* r q12

q11 1->1 r q11 //движение головки в конец ответа

q11 λ->λ ! q11 //остановка машины

q12 1->1 r q12 //движение в конец скопированного слова

q12 0->0 r q12 //движение в конец скопированного слова

q12 λ->\* r q13 //установка разделителя

q13 λ->\* l q1 //установка разделителя

//алгоритм перевода из двоичной СС в натуральную (основной алгоритм)

q1 1->0 r q14 // вычет из единицы + переход в состояние для написания ответа

q1 0->1 l q17 //вычет из нуля + переход к вычету из следующего разряда

q13 1->- r q13 //замена разделителя

q13 \*->→ r q18 //замена разделителя

q14 1->1 r q14 //установка головки в конец ответа

q14 0->0 r q14 //установка головки в конец ответа

q14 \*->\* r q14 //установка головки в конец ответа

q14 λ->1 l q15 //пишем единицу и переходим в состояние для возврата к последнему символу

q15 1->1 l q15 //возвращаемся к последнему символу скопированного слова

q15 0->0 l q15 //возвращаемся к последнему символу скопированного слова

q15 \*->\* l q1 //переход в состояние для вычитания

q17 1->0 r q14 //вычет из следующего разряда

q17 0->1 l q17 //вычет из следующего разряда (опять 0-1 => переходим к еще одному разряду)

q17 \*->- r q13 //замена разделителя

q18 \*->λ r q11 //замена разделителя

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

00, ,\*,00 //ставим разделитель

00,\*,>,01 //двигаемся вправо, чтобы поставить ещё один разделитель

01, ,\*,02 //ставим еще один разделитель

01,\*,<,01 //движение к последнему символу скопированного слова

01,1,0,14 //вычет единицы из единицы

01,0,1,16 //вычет единицы из нуля + переход в состояние для занимания из следующего разряда

02,\*,<,02 //движение в начало слова после установки разделителей

02,1,<,02 //движение в начало слова после установки разделителей

02,0,<,02 //движение в начало слова после установки разделителей

02, ,>,03 //движение на первый символ слова + смена состояния

03,1, ,04 //стираем единицу и запоминаем ее с помощью состояния

03,0, ,07 //стираем ноль и запоминаем его с помощью состояния

03,\*, ,10 //завершили копирование слова, убираем разделитель

04,1,>,03 //только что поставили 1 => двигаемся вправо (нам не нужно с ней ничего делать)

04, ,>,05 //смена состояния

05,1,>,05 //установка головки после разделителей и ответа (чтобы перенести 1)

05,0,>,05 //установка головки после разделителей и ответа (чтобы перенести 1)

05,\*,>,05 //установка головки после разделителей и ответа (чтобы перенести 1)

05, ,1,06 //пишем единицу + меняем состояние (для ее возврата)

06,1,<,06 //передвижение головки обратно (на место стертого символа)

06,0,<,06 //передвижение головки обратно (на место стертого символа)

06,\*,<,06 //передвижение головки обратно (на место стертого символа)

06, ,1,04 //пишем единицу (возвращаем её на место) + меняем состояние

07,1,>,07 //установка головки после разделителей и ответа

07, ,>,07 //установка головки после разделителей и ответа

07,0,>,07 //установка головки после разделителей и ответа

07,\*,>,08 //смена состояния

08,1,>,08 //установка головки после разделителей и ответа (чтобы перенести 0)

08,0,>,08 //установка головки после разделителей и ответа (чтобы перенести 0)

08, ,0,09 //пишем ноль + меняем состояние (для его возврата)

08,\*,>,08 //установка головки после разделителей и ответа (чтобы перенести 0)

09,1,<,09 //передвижение головки обратно (на место стертого символа)

09,0,<,09 //передвижение головки обратно (на место стертого символа)

09,\*,<,09 //передвижение головки обратно (на место стертого символа)

09, ,0,04 //пишем ноль (возвращаем его на место) + меняем состояние

04,0,>,03 //только что поставили 0 => двигаемся вправо (нам не нужно с ним ничего делать)

10, ,>,11 //движение в конец скопированного слова

11,\*,>,12 //движение в конец скопированного слова

11,1,>,11 //движение в конец ответа

11, ,#,11 // остановка МТ

12,1,>,12 //движение в конец скопированного слова

12,0,>,12 //движение в конец скопированного слова

12, ,\*,12 //установка разделителя в конце скопированного слова

12,\*,>,13 //установка разделителя в конце скопированного слова

13, ,\*,01 //установка разделителя в конце скопированного слова

13,-,>,13 //убираем разделитель

13,1,-,13 //убираем разделитель

13,\*,},18 //убираем разделитель

14,1,>,14 //установка головки после разделителя

14,0,>,14 //установка головки после разделителя

14,\*,>,14 //установка головки после разделителя

14, ,1,15 //пишем единицу + переходим в состояние для возвращения к слову

15,1,<,15 //возвращаемся к скопированному слову

15,\*,<,01 //возвращаемся к скопированному слову + смена состояния для дальнейшего вычета

16,1,<,17 //двигаемся влево (к следующему разряду)

17,1,0,14 //вычет из разряда + переход в состояние для движения за разделитель

17,0,1,16 //вычет из разряда + переход к другому разряду

17,\*,-,13 //убираем разделитель

18,},>,18 //убираем разделитель

18,\*, ,19 //убираем разделитель

19, ,>,11//установка головки в конец ответа

Допущен к выполнению работы. Подпись преподавателя

1. **Распечатка протокола** (подклеить листинг окончательного варианта программы с текстовыми примерами, подписанный преподавателем)

**isitmuse@isitmuse**:**~**$ cd turun

**isitmuse@isitmuse**:**~/turun**$ cat head.txt

\_.\_ \_,-'""`-.\_

(,-.`.\_,'( |\`-/|

`-.-' \ )-`( , o o)

`- \`\_`"'-

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|||ЛАБОРАТОРНАЯ РАБОТА №5 |||

|||||ПРОГРАММИРОВАНИЕ МТ|||||

|||ВЫПОЛНИЛ СТУДЕНТ ГРУППЫ|||

|||М8О-105Б-21 КОЗЛОВ ЕГОР|||

\\\\\\\\\\\\\\|//////////////

**isitmuse@isitmuse**:**~/turun**$ cat > test

1010

^

11001

^

1111

^

10001

^

**isitmuse@isitmuse**:**~/turun**$ ls

head.txt MT.tu test **turun**

**isitmuse@isitmuse**:**~/turun**$ ./turun MT.tu test

00, ,\*,00 04,1,>,03 07,0,>,07 11,\*,>,12 14,0,>,14

00,\*,>,01 04, ,>,05 07,\*,>,08 11,1,>,11 14,\*,>,14

01, ,\*,02 05,1,>,05 08,1,>,08 11, ,#,11 14, ,1,15

01,\*,<,01 05,0,>,05 08,0,>,08 12,1,>,12 15,1,<,15

01,1,0,14 05,\*,>,05 08, ,0,09 12,0,>,12 15,\*,<,01

01,0,1,16 05, ,1,06 08,\*,>,08 12, ,\*,12 16,1,<,17

02,\*,<,02 06,1,<,06 09,1,<,09 12,\*,>,13 17,1,0,14

02,1,<,02 06,0,<,06 09,0,<,09 13, ,\*,01 17,0,1,16

02,0,<,02 06,\*,<,06 09,\*,<,09 13,-,>,13 17,\*,-,13

02, ,>,03 06, ,1,04 09, ,0,04 13,1,-,13 18,},>,18

03,1, ,04 07,1,>,07 04,0,>,03 13,\*,},18 18,\*, ,19

03,0, ,07 07, ,>,07 10, ,>,11 14,1,>,14 19, ,>,11

03,\*, ,10

1 =>

1\* =>

1\*\* =>

\*\* =>

\*\*1 =>

1\*\*1 =>

1 \*1 =>

1 \*1\* =>

1 \*1\*\* =>

1 \*0\*\* =>

1 \*0\*\*1 =>

1 \*1\*\*1 =>

1 -1\*\*1 =>

1 --\*\*1 =>

1 --}\*1 =>

1 --} 1 =>

1 --} 1

Machine stopped successfully

10 =>

10\* =>

10\*\* =>

0\*\* =>

0\*\*1 =>

10\*\*1 =>

1 \*\*1 =>

1 \*\*10 =>

10\*\*10 =>

10 \*10 =>

10 \*10\* =>

10 \*10\*\* =>

10 \*11\*\* =>

10 \*01\*\* =>

10 \*01\*\*1 =>

10 \*00\*\*1 =>

10 \*00\*\*11 =>

10 \*01\*\*11 =>

10 \*11\*\*11 =>

10 -11\*\*11 =>

10 --1\*\*11 =>

10 ---\*\*11 =>

10 ---}\*11 =>

10 ---} 11 =>

10 ---} 11

Machine stopped successfully

101 =>

101\* =>

101\*\* =>

01\*\* =>

01\*\*1 =>

101\*\*1 =>

1 1\*\*1 =>

1 1\*\*10 =>

101\*\*10 =>

10 \*\*10 =>

10 \*\*101 =>

101\*\*101 =>

101 \*101 =>

101 \*101\* =>

101 \*101\*\* =>

101 \*100\*\* =>

101 \*100\*\*1 =>

101 \*101\*\*1 =>

101 \*111\*\*1 =>

101 \*011\*\*1 =>

101 \*011\*\*11 =>

101 \*010\*\*11 =>

101 \*010\*\*111 =>

101 \*011\*\*111 =>

101 \*001\*\*111 =>

101 \*001\*\*1111 =>

101 \*000\*\*1111 =>

101 \*000\*\*11111 =>

101 \*001\*\*11111 =>

101 \*011\*\*11111 =>

101 \*111\*\*11111 =>

101 -111\*\*11111 =>

101 --11\*\*11111 =>

101 ---1\*\*11111 =>

101 ----\*\*11111 =>

101 ----}\*11111 =>

101 ----} 11111 =>

101 ----} 11111

Machine stopped successfully

111 =>

111\* =>

111\*\* =>

11\*\* =>

11\*\*1 =>

111\*\*1 =>

1 1\*\*1 =>

1 1\*\*11 =>

111\*\*11 =>

11 \*\*11 =>

11 \*\*111 =>

111\*\*111 =>

111 \*111 =>

111 \*111\* =>

111 \*111\*\* =>

111 \*110\*\* =>

111 \*110\*\*1 =>

111 \*111\*\*1 =>

111 \*101\*\*1 =>

111 \*101\*\*11 =>

111 \*100\*\*11 =>

111 \*100\*\*111 =>

111 \*101\*\*111 =>

111 \*111\*\*111 =>

111 \*011\*\*111 =>

111 \*011\*\*1111 =>

111 \*010\*\*1111 =>

111 \*010\*\*11111 =>

111 \*011\*\*11111 =>

111 \*001\*\*11111 =>

111 \*001\*\*111111 =>

111 \*000\*\*111111 =>

111 \*000\*\*1111111 =>

111 \*001\*\*1111111 =>

111 \*011\*\*1111111 =>

111 \*111\*\*1111111 =>

111 -111\*\*1111111 =>

111 --11\*\*1111111 =>

111 ---1\*\*1111111 =>

111 ----\*\*1111111 =>

111 ----}\*1111111 =>

111 ----} 1111111 =>

111 ----} 1111111

Machine stopped successfully

1. **Дневник отладки** должен содержать дату и время сеансов отладки, и основные ошибки (ошибки в сценарии и программе, не стандартные операции) 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 | 17.10.2021 | 10:32 | Нашел много лишних состояний в МТ (можно обойтись и без них) | Переписал МТ без этих состояний |  |
| 2\* | дом | 17.10.2021 | 10:43 | turun не воспринимает символ → (из-за него программа ломается) | Отказ от использования символа → | Заменил символ на похожий(}) |

1. Замечание автора по существу работы:

Машина Тьюринга в пятерках отличается от машины Тьюринга в четверках. Машина Тьюринга, написанная в пятерках будет короче, так как при программировании МТ в пятерках можно совершить два действия одновременно (замену символа и перемещение), что позволяет сократить количество состояний.

*\*к символам → и }*

*я хотел сделать что-то похожее на стрелочку(знак перевода из одной сс в другую), чтобы выглядело красиво. Если бы я использовал символ →, то стрелочка выглядела бы так: ------→, но так как я использовал символ }, то она выглядит так: -------}.*

1. Выводы:

В ходе лабораторной работы я научился программировать машину Тьюринга в четверках и пятерках и получил навыки для выполнения лабораторных работ по машине Тьюринга.

Подпись студента