**Отчет по лабораторной работе № 5** по курсу по курсу “Фундаментальная информатика”

Студент группы [**М8О-107Б-20**](https://vk.com/im?sel=c51) Чекменев Вячеслав Алексеевич, № по списку 25

Контакты e-mail: [chekmenev031@gmail.com](mailto:chekmenev031@gmail.com), telegram: @suraba03

Работа выполнена: «1» октября 2020 г.

Преподаватель: каф. 806 Найденов Иван Евгеньевич

Отчет сдан « » \_\_\_\_\_\_\_\_\_20 \_\_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. **Тема:** программирование машин Тьюринга

1. **Цель работы:** составить машину тьюринга, выполняюшую определенные действия над словами
2. **Задание** (*вариант №* 45 )**:** перевод из троичной СС в девятеричную.
3. **Оборудование** (студента):

Процессор *Intel Core i5-8265U* с ОП 7851 Мб, НМД 256 Гб. Монитор *1920x1080*

1. **Программное обеспечение (студента):**

Операционная система семейства UNIX: linux, наименование: manjaro, версия: 20.1 Mikah

интерпретатор команд: bash, версия: 5.0.18.

текствый редактор: gedit

Утилиты операционной системы: –

Прикладные системы и программы: эмулятор МТ в четверках

Местонахождение и имена файлов программ и данных –

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

Идея алгоритма: проверить входные данные на четность/нечетность количества цифр. Потом, идя слева направо по числу, переводить по две троичные цифры в одну девятеричную.

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

План:

1) проверка количества цифр троичного числа на четность/нечетность:

**–** пропускаю все нули перед 1 или 2. считываю по две цифры, если первая из них окажется пробелом, то количество цифр (КЦ) – четно, а если вторая – то кол-во цифр нечетно.

– создаю две ветви программы: для четного и нечетного кол-ва цифр.

2) ветвь для четного кол-ва цифр:

– иду слева направо, пропускаю нули перед первой цифрами 1 или 2

– начинаю читать по две цифры, стирать, идти циклом до конца всех слов, там ставить соответствующее число в девятеричной СС. Все эти действия зацикливаю.

3) ветвь для нечетного кол-ва цифр:

– иду слева направо, пропускаю нули перед первой цифрами 1 или 2

– вижу первую цифру 1 или 2, стираю ее, иду до конца всех слов ставлю там соответствующее девятеричное число.

– начинаю читать по две цифры, стирать, идти циклом до конца всех слов, там ставить соответствующее число в девятеричной СС. Все эти действия зацикливаю.

4) если на вводе только нули, вывожу один(количетсво) ноль.

Тесты:

– тесты для полной программы:

|  |  |  |
| --- | --- | --- |
| **Входные данные** | **Выходные данные** | **Описание тестируемого случая** |
| 101112 | 345 | проверяет работу с четным КЦ (кол-вом цифр) |
| 10121 | 117 | проверяет работу с нечетным КЦ |
| 000101112 | 345 | проверяет работу с четным КЦ с ведущими нулями |
| 00010121 | 117 | проверяет работу с нечетным КЦ с ведущими нулями |
| 0 | 0 | проверяет работу с одним нулем |
| 0000 | 0 | проверяет работу с n >= 1 нулями |
| 1 | 1 | проверяет работу с одним числом != 0 |

тесты для проверки на четность/нечетность КЦ:

|  |  |  |
| --- | --- | --- |
| **Входные данные** | **Выходные данные** | **Описание тестируемого случая** |
| 101112 | 0 | проверяет на работу с четным КЦ |
| 10121 | 1 | Проверяет на работу с нечетным КЦ |

**8. Распечатка протокола** (подклеить листинг окончательного варианта программы с тестовыми примерами,подписанныйпреподавателем).

//Полная программа начало

00, ,<,00001

00001,0,<,00001

00001,1,<,00001 // иду в начало ленты

00001,2,<,00001

00001, ,>,00802

00802,0,>,00802

00802, ,>,01802

01802, ,0,02802

02802,0,>,03802

03802, , ,03802

00802,1,<,09002

00802,2,<,09002

09002,0,>,09003

09002, ,>,09003

09003, ,>,00031 // это для случая четности

09003,0, ,00032

09003,1, ,00033 // стираю первую цифру

09003,2, ,00034

00031, ,<,00041 // встаю на пробел рядом справа от числа

00041, ,<,00010

// begin (четные)

00010,0,<,00010

00010,1,<,00010

00010,2,<,00010

00010, ,>,00290

00290,0,>,00290

00290, ,>,00490

00490, ,0,00590

00590,0,>,00690

00690, , ,00690

00290,1,<,00390

00290,2,<,00390

00390, ,>,00020

00390,0,>,00020

00020,0, ,00310

00020,1, ,00320 // 1

00020,2, ,00330

00310, ,>,00410

00320, ,>,00420 // 2

00330, ,>,00430

00410,0, ,05110

00410,1, ,05120

00410,2, ,05130

00420,0, ,05210

00420,1, ,05220 // 3

00420,2, ,05230

00430,0, ,05310

00430,1, ,05320

00430,2, ,05330

05110, ,>,06110

05120, ,>,06120

05130, ,>,06130

05210, ,>,06210

05220, ,>,06220 // 4

05230, ,>,06230

05310, ,>,06310

05320, ,>,06320

05330, ,>,06330

06110,0,>,06110

06110,1,>,06110 // 5

06110,2,>,06110

06120,0,>,06120

06120,1,>,06120 // 5

06120,2,>,06120

06130,0,>,06130

06130,1,>,06130 // 5

06130,2,>,06130

06210,0,>,06210

06210,1,>,06210 // 5

06210,2,>,06210

06220,0,>,06220

06220,1,>,06220 // 5

06220,2,>,06220

06230,0,>,06230

06230,1,>,06230 // 5

06230,2,>,06230

06310,0,>,06310

06310,1,>,06310 // 5

06310,2,>,06310

06320,0,>,06320

06320,1,>,06320 // 5

06320,2,>,06320

06330,0,>,06330

06330,1,>,06330 // 5

06330,2,>,06330

06110, ,>,07110

06120, ,>,07120

06130, ,>,07130

06210, ,>,07210

06220, ,>,07220 // 6

06230, ,>,07230

06310, ,>,07310

06320, ,>,07320

06330, ,>,07330

07110,0,>,07110

07110,1,>,07110

07110,2,>,07110

07110,3,>,07110

07110,4,>,07110 // 7

07110,5,>,07110

07110,6,>,07110

07110,7,>,07110

07110,8,>,07110

07120,0,>,07120

07120,1,>,07120

07120,2,>,07120

07120,3,>,07120

07120,4,>,07120 // 7

07120,5,>,07120

07120,6,>,07120

07120,7,>,07120

07120,8,>,07120

07130,0,>,07130

07130,1,>,07130

07130,2,>,07130

07130,3,>,07130

07130,4,>,07130 // 7

07130,5,>,07130

07130,6,>,07130

07130,7,>,07130

07130,8,>,07130

07210,0,>,07210

07210,1,>,07210

07210,2,>,07210

07210,3,>,07210

07210,4,>,07210 // 7

07210,5,>,07210

07210,6,>,07210

07210,7,>,07210

07210,8,>,07210

07220,0,>,07220

07220,1,>,07220

07220,2,>,07220

07220,3,>,07220

07220,4,>,07220 // 7

07220,5,>,07220

07220,6,>,07220

07220,7,>,07220

07220,8,>,07220

07230,0,>,07230

07230,1,>,07230

07230,2,>,07230

07230,3,>,07230

07230,4,>,07230 // 7

07230,5,>,07230

07230,6,>,07230

07230,7,>,07230

07230,8,>,07230

07310,0,>,07310

07310,1,>,07310

07310,2,>,07310

07310,3,>,07310

07310,4,>,07310 // 7

07310,5,>,07310

07310,6,>,07310

07310,7,>,07310

07310,8,>,07310

07320,0,>,07320

07320,1,>,07320

07320,2,>,07320

07320,3,>,07320

07320,4,>,07320 // 7

07320,5,>,07320

07320,6,>,07320

07320,7,>,07320

07320,8,>,07320

07330,0,>,07330

07330,1,>,07330

07330,2,>,07330

07330,3,>,07330

07330,4,>,07330 // 7

07330,5,>,07330

07330,6,>,07330

07330,7,>,07330

07330,8,>,07330

07110, ,0,08110

07120, ,1,08120

07130, ,2,08130

07210, ,3,08210

07220, ,4,08220 // 8

07230, ,5,08230

07310, ,6,08310

07320, ,7,08320

07330, ,8,08330

08110,0,<,08110

08110,1,<,08110

08110,2,<,08110

08110,3,<,08110

08110,4,<,08110 // 9

08110,5,<,08110

08110,6,<,08110

08110,7,<,08110

08110,8,<,08110

08120,0,<,08120

08120,1,<,08120

08120,2,<,08120

08120,3,<,08120

08120,4,<,08120 // 9

08120,5,<,08120

08120,6,<,08120

08120,7,<,08120

08120,8,<,08120

08130,0,<,08130

08130,1,<,08130

08130,2,<,08130

08130,3,<,08130

08130,4,<,08130 // 9

08130,5,<,08130

08130,6,<,08130

08130,7,<,08130

08130,8,<,08130

08210,0,<,08210

08210,1,<,08210

08210,2,<,08210

08210,3,<,08210

08210,4,<,08210 // 9

08210,5,<,08210

08210,6,<,08210

08210,7,<,08210

08210,8,<,08210

08220,0,<,08220

08220,1,<,08220

08220,2,<,08220

08220,3,<,08220

08220,4,<,08220 // 9

08220,5,<,08220

08220,6,<,08220

08220,7,<,08220

08220,8,<,08220

08230,0,<,08230

08230,1,<,08230

08230,2,<,08230

08230,3,<,08230

08230,4,<,08230 // 9

08230,5,<,08230

08230,6,<,08230

08230,7,<,08230

08230,8,<,08230

08310,0,<,08310

08310,1,<,08310

08310,2,<,08310

08310,3,<,08310

08310,4,<,08310 // 9

08310,5,<,08310

08310,6,<,08310

08310,7,<,08310

08310,8,<,08310

08320,0,<,08320

08320,1,<,08320

08320,2,<,08320

08320,3,<,08320

08320,4,<,08320 // 9

08320,5,<,08320

08320,6,<,08320

08320,7,<,08320

08320,8,<,08320

08330,0,<,08330

08330,1,<,08330

08330,2,<,08330

08330,3,<,08330

08330,4,<,08330 // 9

08330,5,<,08330

08330,6,<,08330

08330,7,<,08330

08330,8,<,08330

08110, ,<,09110

08120, ,<,09120

08130, ,<,09130

08210, ,<,09210

08220, ,<,09220 // 10

08230, ,<,09230

08310, ,<,09310

08320, ,<,09320

08330, ,<,09330

09110,0,<,09110

09110,1,<,09110 // 11

09110,2,<,09110

09120,0,<,09120

09120,1,<,09120 // 11

09120,2,<,09120

09130,0,<,09130

09130,1,<,09130 // 11

09130,2,<,09130

09210,0,<,09210

09210,1,<,09210 // 11

09210,2,<,09210

09220,0,<,09220

09220,1,<,09220 // 11

09220,2,<,09220

09230,0,<,09230

09230,1,<,09230 // 11

09230,2,<,09230

09310,0,<,09310

09310,1,<,09310 // 11

09310,2,<,09310

09320,0,<,09320

09320,1,<,09320 // 11

09320,2,<,09320

09330,0,<,09330

09330,1,<,09330 // 11

09330,2,<,09330

09110, ,<,10110

09120, ,<,10120

09130, ,<,10130

09210, ,<,10210

09220, ,<,10220 // 12

09230, ,<,10230

09310, ,<,10310

09320, ,<,10320

09330, ,<,10330

10110, ,0,11110

10120, ,0,11120 // 13

10130, ,0,11130

10210, ,1,11210

10220, ,1,11220 // 13

10230, ,1,11230

10310, ,2,11310

10320, ,2,11320 // 13

10330, ,2,11330

11110,0,>,12110

11120,0,>,12120 // 14

11130,0,>,12130

11210,1,>,12210

11220,1,>,12220 // 14

11230,1,>,12230

11310,2,>,12310

11320,2,>,12320 // 14

11330,2,>,12330

12110, ,0,13110

12120, ,1,13120 // 15

12130, ,2,13130

12210, ,0,13210

12220, ,1,13220 // 15

12230, ,2,13230

12310, ,0,13310

12320, ,1,13320 // 15

12330, ,2,13330

13110,0,>,00020

13120,1,>,00020

13130,2,>,00020

13210,0,>,00020

13220,1,>,00020 // 16, цикл замкнут

13230,2,>,00020

13310,0,>,00020

13320,1,>,00020

13330,2,>,00020

00020, ,>,14000

14000,0,>,14000

14000,1,>,14000

14000,2,>,14000

14000,3,>,14000

14000,4,>,14000

14000,5,>,14000

14000,6,>,14000

14000,7,>,14000

14000,8,>,14000

14000, , ,14000

// end (четные)

00032, ,>,00042

00033, ,>,00043 // перехожу ко второму числу

00034, ,>,00044

00042, ,<,00521

00043, ,<,00531 // это для случая нечетности

00044, ,<,00541

00042,0, ,00522

00042,1, ,00523 // стираю вторую цифру

00042,2, ,00524

00043,0, ,00532

00043,1, ,00533 // стираю вторую цифру

00043,2, ,00534

00044,0, ,00542

00044,1, ,00543 // стираю вторую цифру

00044,2, ,00544

00521, ,0,00621

00531, ,1,00631 // возвращаю последнюю цифру

00541, ,2,00641

00621,0,>,10621

00631,1,>,10631 // встаю на первый после слова пробел

00641,2,>,10641

10621, ,<,60001

10631, ,<,60001 // иду в программу для нечетного кол-ва

10641, ,<,60001

// begin (нечетные)

60001,0,<,60001

60001,1,<,60001

60001,2,<,60001

60001, ,>,80002

80002,0,>,80002

80002, ,>,82002

82002, ,0,83002

83002,0,>,84002

84002, , ,84002

80002,1,<,81002

80002,2,<,81002

81002,0,>,50002

81002, ,>,50002

//

50002,1, ,50031

50002,2, ,50032

50031, ,>,50041

50032, ,>,50042

50041,0,>,50041

50041,1,>,50041

50041,2,>,50041

50042,0,>,50042

50042,1,>,50042

50042,2,>,50042

50041, ,>,50051

50042, ,>,50052

50051, ,1,50061

50052, ,2,50062

50061,1,<,50071

50062,2,<,50072

50071, ,<,50081

50072, ,<,50082

50081,0,<,50081

50081,1,<,50081

50081,2,<,50081

50082,0,<,50082

50082,1,<,50082

50082,2,<,50082

50081, ,1,50091

50082, ,2,50092

50091,1,>,60002

50092,2,>,60002

//

60002,0, ,60031

60002,1, ,60032 // 1

60002,2, ,60033

60031, ,>,60041

60032, ,>,60042 // 2

60033, ,>,60043

60041,0, ,60511

60041,1, ,60512

60041,2, ,60513

60042,0, ,60521

60042,1, ,60522 // 3

60042,2, ,60523

60043,0, ,60531

60043,1, ,60532

60043,2, ,60533

60511, ,>,60611

60512, ,>,60612

60513, ,>,60613

60521, ,>,60621

60522, ,>,60622 // 4

60523, ,>,60623

60531, ,>,60631

60532, ,>,60632

60533, ,>,60633

60611,0,>,60611

60611,1,>,60611 // 5

60611,2,>,60611

60612,0,>,60612

60612,1,>,60612 // 5

60612,2,>,60612

60613,0,>,60613

60613,1,>,60613 // 5

60613,2,>,60613

60621,0,>,60621

60621,1,>,60621 // 5

60621,2,>,60621

60622,0,>,60622

60622,1,>,60622 // 5

60622,2,>,60622

60623,0,>,60623

60623,1,>,60623 // 5

60623,2,>,60623

60631,0,>,60631

60631,1,>,60631 // 5

60631,2,>,60631

60632,0,>,60632

60632,1,>,60632 // 5

60632,2,>,60632

60633,0,>,60633

60633,1,>,60633 // 5

60633,2,>,60633

60611, ,>,60711

60612, ,>,60712

60613, ,>,60713

60621, ,>,60721

60622, ,>,60722 // 6

60623, ,>,60723

60631, ,>,60731

60632, ,>,60732

60633, ,>,60733

60711,0,>,60711

60711,1,>,60711

60711,2,>,60711

60711,3,>,60711

60711,4,>,60711 // 7

60711,5,>,60711

60711,6,>,60711

60711,7,>,60711

60711,8,>,60711

60712,0,>,60712

60712,1,>,60712

60712,2,>,60712

60712,3,>,60712

60712,4,>,60712 // 7

60712,5,>,60712

60712,6,>,60712

60712,7,>,60712

60712,8,>,60712

60713,0,>,60713

60713,1,>,60713

60713,2,>,60713

60713,3,>,60713

60713,4,>,60713 // 7

60713,5,>,60713

60713,6,>,60713

60713,7,>,60713

60713,8,>,60713

60721,0,>,60721

60721,1,>,60721

60721,2,>,60721

60721,3,>,60721

60721,4,>,60721 // 7

60721,5,>,60721

60721,6,>,60721

60721,7,>,60721

60721,8,>,60721

60722,0,>,60722

60722,1,>,60722

60722,2,>,60722

60722,3,>,60722

60722,4,>,60722 // 7

60722,5,>,60722

60722,6,>,60722

60722,7,>,60722

60722,8,>,60722

60723,0,>,60723

60723,1,>,60723

60723,2,>,60723

60723,3,>,60723

60723,4,>,60723 // 7

60723,5,>,60723

60723,6,>,60723

60723,7,>,60723

60723,8,>,60723

60731,0,>,60731

60731,1,>,60731

60731,2,>,60731

60731,3,>,60731

60731,4,>,60731 // 7

60731,5,>,60731

60731,6,>,60731

60731,7,>,60731

60731,8,>,60731

60732,0,>,60732

60732,1,>,60732

60732,2,>,60732

60732,3,>,60732

60732,4,>,60732 // 7

60732,5,>,60732

60732,6,>,60732

60732,7,>,60732

60732,8,>,60732

60733,0,>,60733

60733,1,>,60733

60733,2,>,60733

60733,3,>,60733

60733,4,>,60733 // 7

60733,5,>,60733

60733,6,>,60733

60733,7,>,60733

60733,8,>,60733

60711, ,0,60811

60712, ,1,60812

60713, ,2,60813

60721, ,3,60821

60722, ,4,60822 // 8

60723, ,5,60823

60731, ,6,60831

60732, ,7,60832

60733, ,8,60833

60811,0,<,60811

60811,1,<,60811

60811,2,<,60811

60811,3,<,60811

60811,4,<,60811 // 9

60811,5,<,60811

60811,6,<,60811

60811,7,<,60811

60811,8,<,60811

60812,0,<,60812

60812,1,<,60812

60812,2,<,60812

60812,3,<,60812

60812,4,<,60812 // 9

60812,5,<,60812

60812,6,<,60812

60812,7,<,60812

60812,8,<,60812

60813,0,<,60813

60813,1,<,60813

60813,2,<,60813

60813,3,<,60813

60813,4,<,60813 // 9

60813,5,<,60813

60813,6,<,60813

60813,7,<,60813

60813,8,<,60813

60821,0,<,60821

60821,1,<,60821

60821,2,<,60821

60821,3,<,60821

60821,4,<,60821 // 9

60821,5,<,60821

60821,6,<,60821

60821,7,<,60821

60821,8,<,60821

60822,0,<,60822

60822,1,<,60822

60822,2,<,60822

60822,3,<,60822

60822,4,<,60822 // 9

60822,5,<,60822

60822,6,<,60822

60822,7,<,60822

60822,8,<,60822

60823,0,<,60823

60823,1,<,60823

60823,2,<,60823

60823,3,<,60823

60823,4,<,60823 // 9

60823,5,<,60823

60823,6,<,60823

60823,7,<,60823

60823,8,<,60823

60831,0,<,60831

60831,1,<,60831

60831,2,<,60831

60831,3,<,60831

60831,4,<,60831 // 9

60831,5,<,60831

60831,6,<,60831

60831,7,<,60831

60831,8,<,60831

60832,0,<,60832

60832,1,<,60832

60832,2,<,60832

60832,3,<,60832

60832,4,<,60832 // 9

60832,5,<,60832

60832,6,<,60832

60832,7,<,60832

60832,8,<,60832

60833,0,<,60833

60833,1,<,60833

60833,2,<,60833

60833,3,<,60833

60833,4,<,60833 // 9

60833,5,<,60833

60833,6,<,60833

60833,7,<,60833

60833,8,<,60833

60811, ,<,60911

60812, ,<,60912

60813, ,<,60913

60821, ,<,60921

60822, ,<,60922 // 10

60823, ,<,60923

60831, ,<,60931

60832, ,<,60932

60833, ,<,60933

60911,0,<,60911

60911,1,<,60911 // 11

60911,2,<,60911

60912,0,<,60912

60912,1,<,60912 // 11

60912,2,<,60912

60913,0,<,60913

60913,1,<,60913 // 11

60913,2,<,60913

60921,0,<,60921

60921,1,<,60921 // 11

60921,2,<,60921

60922,0,<,60922

60922,1,<,60922 // 11

60922,2,<,60922

60923,0,<,60923

60923,1,<,60923 // 11

60923,2,<,60923

60931,0,<,60931

60931,1,<,60931 // 11

60931,2,<,60931

60932,0,<,60932

60932,1,<,60932 // 11

60932,2,<,60932

60933,0,<,60933

60933,1,<,60933 // 11

60933,2,<,60933

60911, ,<,61011

60912, ,<,61012

60913, ,<,61013

60921, ,<,61021

60922, ,<,61022 // 12

60923, ,<,61023

60931, ,<,61031

60932, ,<,61032

60933, ,<,61033

61011, ,0,61111

61012, ,0,61112 // 13

61013, ,0,61113

61021, ,1,61121

61022, ,1,61122 // 13

61023, ,1,61123

61031, ,2,61131

61032, ,2,61132 // 13

61033, ,2,61133

61111,0,>,61211

61112,0,>,61212 // 14

61113,0,>,61213

61121,1,>,61221

61122,1,>,61222 // 14

61123,1,>,61223

61131,2,>,61231

61132,2,>,61232 // 14

61133,2,>,61233

61211, ,0,61311

61212, ,1,61312 // 15

61213, ,2,61313

61221, ,0,61321

61222, ,1,61322 // 15

61223, ,2,61323

61231, ,0,61331

61232, ,1,61332 // 15

61233, ,2,61333

61311,0,>,60002

61312,1,>,60002

61313,2,>,60002

61321,0,>,60002

61322,1,>,60002 // 16, цикл замкнут

61323,2,>,60002

61331,0,>,60002

61332,1,>,60002

61333,2,>,60002

60002, ,>,61400

61400,0,>,61400

61400,1,>,61400

61400,2,>,61400

61400,3,>,61400

61400,4,>,61400

61400,5,>,61400

61400,6,>,61400

61400,7,>,61400

61400,8,>,61400

61400, , ,61400

// end (нечетные)

00522, ,>,00622

00523, ,>,00623 // иду вправо

00524, ,>,00624

00532, ,>,00632

00533, ,>,00633 // иду вправо

00534, ,>,00634

00542, ,>,00642

00543, ,>,00643 // иду вправо

00544, ,>,00644

00622,0,>,00622

00622,1,>,00622 // иду циклом до пробела вправо

00622,2,>,00622

00623,0,>,00623

00623,1,>,00623 // иду циклом до пробела вправо

00623,2,>,00623

00624,0,>,00624

00624,1,>,00624 // иду циклом до пробела вправо

00624,2,>,00624

00632,0,>,00632

00632,1,>,00632 // иду циклом до пробела вправо

00632,2,>,00632

00633,0,>,00633

00633,1,>,00633 // иду циклом до пробела вправо

00633,2,>,00633

00634,0,>,00634

00634,1,>,00634 // иду циклом до пробела вправо

00634,2,>,00634

00642,0,>,00642

00642,1,>,00642 // иду циклом до пробела вправо

00642,2,>,00642

00643,0,>,00643

00643,1,>,00643 // иду циклом до пробела вправо

00643,2,>,00643

00644,0,>,00644

00644,1,>,00644 // иду циклом до пробела вправо

00644,2,>,00644

00622, ,<,00722

00623, ,<,00723

00624, ,<,00724

00632, ,<,00732

00633, ,<,00733 // иду налево

00634, ,<,00734

00642, ,<,00742

00643, ,<,00743

00644, ,<,00744

00722,0,<,00722

00722,1,<,00722 // иду циклом до пробела влево

00722,2,<,00722

00723,0,<,00723

00723,1,<,00723 // иду циклом до пробела влево

00723,2,<,00723

00724,0,<,00724

00724,1,<,00724 // иду циклом до пробела влево

00724,2,<,00724

00732,0,<,00732

00732,1,<,00732 // иду циклом до пробела влево

00732,2,<,00732

00733,0,<,00733

00733,1,<,00733 // иду циклом до пробела влево

00733,2,<,00733

00734,0,<,00734

00734,1,<,00734 // иду циклом до пробела влево

00734,2,<,00734

00742,0,<,00742

00742,1,<,00742 // иду циклом до пробела влево

00742,2,<,00742

00743,0,<,00743

00743,1,<,00743 // иду циклом до пробела влево

00743,2,<,00743

00744,0,<,00744

00744,1,<,00744 // иду циклом до пробела влево

00744,2,<,00744

00722, ,<,00822

00723, ,<,00823

00724, ,<,00824

00732, ,<,00832

00733, ,<,00833 // иду налево

00734, ,<,00834

00742, ,<,00842

00743, ,<,00843

00744, ,<,00844

00822, ,0,00922

00823, ,0,00923

00824, ,0,00924

00832, ,1,00932

00833, ,1,00933 // возвращаю цифру на место

00834, ,1,00934

00842, ,2,00942

00843, ,2,00943

00844, ,2,00944

00922,0,>,01022

00923,0,>,01023

00924,0,>,01024

00932,1,>,01032

00933,1,>,01033 // иду направо

00934,1,>,01034

00942,2,>,01042

00943,2,>,01043

00944,2,>,01044

01022, ,0,01022

01023, ,1,01023

01024, ,2,01024

01032, ,0,01032

01033, ,1,01033 // возвращаю на место вторую цифру

01034, ,2,01034

01042, ,0,01042

01043, ,1,01043

01044, ,2,01044

01022,0,>,09003

01023,1,>,09003

01024,2,>,09003

01032,0,>,09003

01033,1,>,09003 // конец итерации цикла

01034,2,>,09003

01042,0,>,09003

01043,1,>,09003

01044,2,>,09003

-----------------------------------------------------------// полная программа конец -------------------------------------------

-----------------------// проверка на четность/нечетность с выводом 0/1 соответственно ---------------------------------------------

00000, ,<,00001

00001,0,<,00001

00001,1,<,00001 // иду в начало ленты

00001,2,<,00001

00001, ,>,00002

00002, ,>,00031 // это для случая четности

00031, ,0,00041 // ставлю 0, число имеет четное число цифр

00041,0,>,00412

00412, , ,00412 // заканчиваю программу для четных

00002,0, ,00032

00002,1, ,00033 // стираю первую цифру

00002,2, ,00034

00032, ,>,00042

00033, ,>,00043 // перехожу ко второму числу

00034, ,>,00044

00042, ,<,00521

00043, ,<,00531 // это для случая нечетности

00044, ,<,00541

00042,0, ,00522

00042,1, ,00523 // стираю вторую цифру

00042,2, ,00524

00043,0, ,00532

00043,1, ,00533 // стираю вторую цифру

00043,2, ,00534

00044,0, ,00542

00044,1, ,00543 // стираю вторую цифру

00044,2, ,00544

00521, ,0,00621

00531, ,1,00631

00541, ,2,00641

00621,0,>,10621

00631,1,>,10631

00641,2,>,10641

10621, ,>,20621

10631, ,>,20631

10641, ,>,20641

20621, ,1,30621

20631, ,1,30631

20641, ,1,30641

30621,1,>,40621

30631,1,>,40631

30641,1,>,40641

40621, , ,40621

40631, , ,40631

40641, , ,40641

00522, ,>,00622

00523, ,>,00623 // иду вправо

00524, ,>,00624

00532, ,>,00632

00533, ,>,00633 // иду вправо

00534, ,>,00634

00542, ,>,00642

00543, ,>,00643 // иду вправо

00544, ,>,00644

00621,1,>,06212

06212, , ,06212 // заканчиваю программу для нечетных

00622,0,>,00622

00622,1,>,00622 // иду циклом до пробела вправо

00622,2,>,00622

00623,0,>,00623

00623,1,>,00623 // иду циклом до пробела вправо

00623,2,>,00623

00624,0,>,00624

00624,1,>,00624 // иду циклом до пробела вправо

00624,2,>,00624

00632,0,>,00632

00632,1,>,00632 // иду циклом до пробела вправо

00632,2,>,00632

00633,0,>,00633

00633,1,>,00633 // иду циклом до пробела вправо

00633,2,>,00633

00634,0,>,00634

00634,1,>,00634 // иду циклом до пробела вправо

00634,2,>,00634

00642,0,>,00642

00642,1,>,00642 // иду циклом до пробела вправо

00642,2,>,00642

00643,0,>,00643

00643,1,>,00643 // иду циклом до пробела вправо

00643,2,>,00643

00644,0,>,00644

00644,1,>,00644 // иду циклом до пробела вправо

00644,2,>,00644

00622, ,<,00722

00623, ,<,00723

00624, ,<,00724

00632, ,<,00732

00633, ,<,00733 // иду налево

00634, ,<,00734

00642, ,<,00742

00643, ,<,00743

00644, ,<,00744

00722,0,<,00722

00722,1,<,00722 // иду циклом до пробела влево

00722,2,<,00722

00723,0,<,00723

00723,1,<,00723 // иду циклом до пробела влево

00723,2,<,00723

00724,0,<,00724

00724,1,<,00724 // иду циклом до пробела влево

00724,2,<,00724

00732,0,<,00732

00732,1,<,00732 // иду циклом до пробела влево

00732,2,<,00732

00733,0,<,00733

00733,1,<,00733 // иду циклом до пробела влево

00733,2,<,00733

00734,0,<,00734

00734,1,<,00734 // иду циклом до пробела влево

00734,2,<,00734

00742,0,<,00742

00742,1,<,00742 // иду циклом до пробела влево

00742,2,<,00742

00743,0,<,00743

00743,1,<,00743 // иду циклом до пробела влево

00743,2,<,00743

00744,0,<,00744

00744,1,<,00744 // иду циклом до пробела влево

00744,2,<,00744

00722, ,<,00822

00723, ,<,00823

00724, ,<,00824

00732, ,<,00832

00733, ,<,00833 // иду налево

00734, ,<,00834

00742, ,<,00842

00743, ,<,00843

00744, ,<,00844

00822, ,0,00922

00823, ,0,00923

00824, ,0,00924

00832, ,1,00932

00833, ,1,00933 // возвращаю цифру на место

00834, ,1,00934

00842, ,2,00942

00843, ,2,00943

00844, ,2,00944

00922,0,>,01022

00923,0,>,01023

00924,0,>,01024

00932,1,>,01032

00933,1,>,01033 // иду направо

00934,1,>,01034

00942,2,>,01042

00943,2,>,01043

00944,2,>,01044

01022, ,0,01022

01023, ,1,01023

01024, ,2,01024

01032, ,0,01032

01033, ,1,01033 // возвращаю на место вторую цифру

01034, ,2,01034

01042, ,0,01042

01043, ,1,01043

01044, ,2,01044

01022,0,>,00002

01023,1,>,00002

01024,2,>,00002

01032,0,>,00002

01033,1,>,00002 // конец итерации цикла

01034,2,>,00002

01042,0,>,00002

01043,1,>,00002

01044,2,>,00002

// проверка КЦ на четность/нечетность с выводом 0/1 соотвественно

**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. | Дата | вре- | Событие | Действие по исправлению | Примечание |
|  | или |  | мя |  |  |  |
|  | дом. |  |  |  |  |  |
| 5 | дом | 23.09.2020 | 01:00 | Написал рабочий алгоритм, который печатает 9ричное число на место 3ичного. Прочитал условие на чекере, оказалось нужно печатать рядом. | Копировать троичное число и выполнять тот же алгоритм, только для скопированного числа. | Нужно сначала читать условие. |
| 5 | маи | 23.09.2020 | 13:00 | Понял, что идея с копированием слишком усложненная | Ищу алгоритм полегче | – |
| 5 | Дом | 25.09.2020 | вечер | Нашел алгоритм попроще: сделать проверку на четность количества цифр в троичном числе. Написать отдельно программу перевода для четного КЦ и нечетного | – | – |
| 5 | дом | 26.09.20 | вечер | Написал выше описанные алгоритмы. Не саработало для числа состоящего из всех нулей. | Исправить, чтобы работало для таких чисел | – |
| 5 | Дом | 27.09.2020 | утро | Исправил, все работает исправно. Загружаю на чекер, не прохожу первый тест. | Попробовать переводить число вместе с ведещими нулями | Не понимаю.. |
| 5 | Дом | 29.09.2020 | вечер | Написал такой алгоритм, все равно не проходит первый тест | Никаких идей | Не понимаю… |
| 5 | Дом | 03.10.2020 | утро | Иван Евгеньевич подсказал исправить нулевое состояние с 00000 на 00, все заработало... | Все исправно |  |

1. **Замечания автора** по существу работы

Условия на тестирующей платформе чекер неполные, сложно понять нюансы задания.

1. **Выводы**

В целом, данная работа мне понравилась, времени на выполнение ушло много, было две неудачные попытки, на третью попытку программа вышла полной и работала на всех моих тестах, однако в чекере не проходила первый тест, потому что первое состояние должно было быть вида 00. После исправления все заработало. Самое интересное и, одновременно, самое сложное в данной ЛР было составление алгоритма, т.к. после написания кода на языках высокого уровня сложно общаться с машиной простейшими командами. Не очень понравилась постребность написания 162-ух почти одинаковых команд для перехода через число (на следующий пробел) в двятеричной СС. Однако несмотря на минусы данной работы она очень хорошо развивает алгоритмическое мышление, которое помогает при написании кода на языках высокого уровня.

Подпись студента: