

# Отчет по лабораторной работе №7 по курсу фундаментальная информатика

Студент группы М8О-104Б-22 Ольхов Максим Алексеевич, № по списку 11

Контакты www, e-mail, icq, skype maksolhov@yandex.ru

Работа выполнена: «20» октября 2022г.

Преподаватель: Потенко М.А. каф. 806

Входной контроль знаний с оценкой \_\_\_\_\_

Отчет сдан «    » \_\_\_\_\_ 201 \_\_\_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_

1. **Тема:** Программирование в алгоритмической модели Маркова
2. **Цель работы:** написать программу алгоритма Маркова
3. **Задание (Индивидуальный вариант):** Циклический сдвиг первого числа влево на число разрядов второго числа (троичные числа).
4. **Оборудование (лабораторное):**  
ЭВМ \_\_\_\_\_, процессор \_\_\_\_\_, имя узла сети \_\_\_\_\_ с ОП \_\_\_\_\_ Мб, НМД \_\_\_\_\_ Мб. Терминал \_\_\_\_\_ адрес \_\_\_\_\_  
\_\_\_\_\_. Принтер \_\_\_\_\_  
Другие устройства \_\_\_\_\_  
*Оборудование ПЭВМ студента, если использовалось:*  
Процессор Intel core i5-7500 3.4 GHz с ОП 8 Гб, НМД 2097152 Мб. Монитор asus  
Другие устройства \_\_\_\_\_
5. **Программное обеспечение (лабораторное):**  
Операционная система семейства \_\_\_\_\_, наименование \_\_\_\_\_ версия \_\_\_\_\_  
интерпретатор команд \_\_\_\_\_ версия \_\_\_\_\_  
Система программирования \_\_\_\_\_ версия \_\_\_\_\_  
Редактор текстов \_\_\_\_\_ версия \_\_\_\_\_  
Утилиты операционной системы \_\_\_\_\_  
Прикладные системы и программы \_\_\_\_\_  
Местонахождение и имена файлов программ и данных \_\_\_\_\_  
*Программное обеспечение ЭВМ студента, если использовалось:*  
Операционная система семейства windows, наименование windows 10 версия 10  
интерпретатор команд \_\_\_\_\_ версия \_\_\_\_\_  
Система программирования \_\_\_\_\_ версия \_\_\_\_\_ Эмулятор машины Тьюринга в четвёрках v2.3 \_\_\_\_\_  
Редактор текстов \_\_\_\_\_ версия \_\_\_\_\_  
Утилиты операционной системы \_\_\_\_\_ Яндекс \_\_\_\_\_  
Прикладные системы и программы markov.exe  
и имена файлов программ и данных на домашнем компьютере \_\_\_\_\_  
\Users\max\Desktop\МАРКОВ\110 строк(испр)3.nma

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

Вся идея заключается в том, что мы с самого начала создаем символ “\*”, чтобы пройти первое число и через “<” перейти в конец второго числа. Как доходим до конца второго числа, мы “\*” на символ “#”, который будет отвечать за вычитание второго числа, после вычитания “#” заменяется на символ “-“, который отвечает за то, чтобы опять вернуть в конец второго числа и одновременно с этим заменять 0 на 2, например 10000# => 1#0000=>02-000=>02222-. Как мы снова дойдем до конца второго числа, мы “-“ заменяем на “!”, чтобы из конца второго числа перейти в начало первого, где мы также заменяем “!” на символы: “;”- за 0 или “^”- за 2 или “?”- за 1(зависит от того, какое число стоит первым), которые будут отвечать за то, чтобы перейти в конец первого числа и заменить последнюю цифру перед знаком “<”. Далее после того, как мы поменяли цифру, мы создаем еще 3 символа, которые будут запоминать цифры “\$” -за 0, “@”- за 1 “№”- за 2. И таким образом мы перемещаем цифры влево по числу. После чего любой из этих трёх символов исчезает в начале первого числа, и цикл повторяется до тех пор, пока слева число не станет равным 0.

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

1) Мы создаем символ “\*” с помощью “->\*”, который будет перемещаться по числу таким образом:

Образец		Замена
*1	→	1*
*2	→	2*
*0	→	0*
*<	→	<*

2) Как дойдем до конца второго числа, мы заменяем “\*” на символ “#”, с помощью “\*->#”, который будет вычитать по единице, после того как мы вычли единицу из числа мы “#” заменяем на символ “-“ (если мы вычитаем только последнюю цифру, то нам не надо менять символ, мы сразу меняем на другой, о котором будет упомянуто в 3 пункте), который отвечает за возвращение, если по возвращению в конец второго числа мы встречаем нули, то эти нули заменяются на 2, таким образом это выглядит:

-0	→	2-
2#0	→	12-
1#0	→	02-
1#	→	0!
2#	→	1!
0#	→	#0

3) Как вычли единицу, мы “-“ или “#” заменяем на символ “!”, который отвечает за то, чтобы перейти в начало первого числа, чтобы там заменить наш символ “!” на символы: “?”-1, “;”-0, “^”-2, всё зависит от того, какая цифра стоит первым и идем в конец числа:

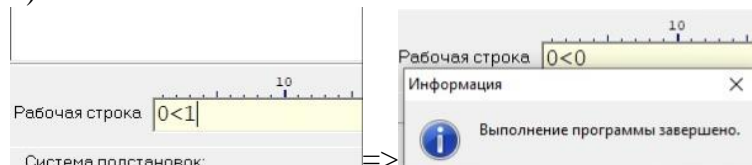
			0	→	
0!	→	!0	!2	→	^2
1!	→	!1	!0	→	;0
2!	→	!2	!1	→	?1
<!	→	!<	<#	→	.<

4) Как дойдем до конца второго числа, мы можем приступать к сдвигу, заменяя 0,1,2 соответственно на “\$”, “@”, “№”, благодаря чему сдвиг возможен. После того, как все цифры были сдвинуты, наши символы исчезают и цикл начинается по новой, пока левое число не станет равным 0. А чтобы правильно программу и не было зацикливаний, нам поможет такая команда: “<#->.<”, в разных программах завершение тоже разное, но в данном случае выглядит именно так. Сам Сдвиг числа выгляди следующим образом:

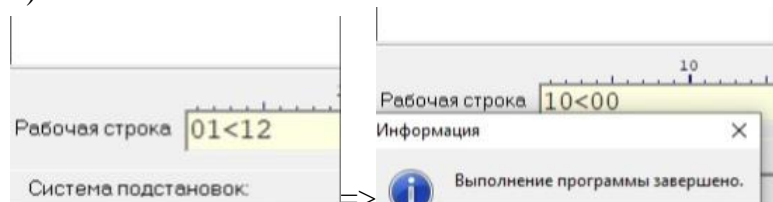
				Образец		Замена
15	?1	→	1?	43	2№2	→ №22
16	?2	→	2?	44	0№2	→ \$22
17	?0	→	0?	45	2№0	→ №20
18	0?<	→	1\$<	46	0\$2	→ \$02
19	1?<	→	1@<	47	2\$1	→ №01
20	2?<	→	1№<	48	0@1	→ \$11
21	^2	→	2^	49	1\$0	→ @00
22	^0	→	0^	50	0@0	→ \$10
23	^1	→	1^	51	1№1	→ @21
24	2^<	→	2№<	52	0№0	→ \$20
25	1^<	→	2@<	53	2\$2	→ №02
26	0^<	→	\$2<	54	2@2	→ №12
27	2@<	→	@2<	55	2@1	→ №11
28	2\$<	→	2\$<	56	1\$2	→ @02
29	;0	→	0;	57	2@0	→ №10
30	;2	→	2;	58	0@2	→ \$12
31	;1	→	1;	59	0@	→ @1
32	0;<	→	0\$<	60	1\$1	→ @01
33	1;<	→	0@<	61	1\$	→ \$1
34	2;<	→	0№<	62	1@	→ @1
35	0@<	→	@0<	63	0\$	→ \$0
36	0№<	→	№0<	64	1^	→ ^1
37	2№<	→	№2<	65	0^	→ ^1
38	2\$0	→	№00	66	2\$	→ \$2
39	1№0	→	@20	67	2@	→ @2
40	0№1	→	\$21	68	1№	→ №1
41	1№2	→	@22	69	№00	→ 00
42	2№1	→	№21	70	№01	→ 01
43	2№2	→	№22	71	№10	→ 10

Тесты:

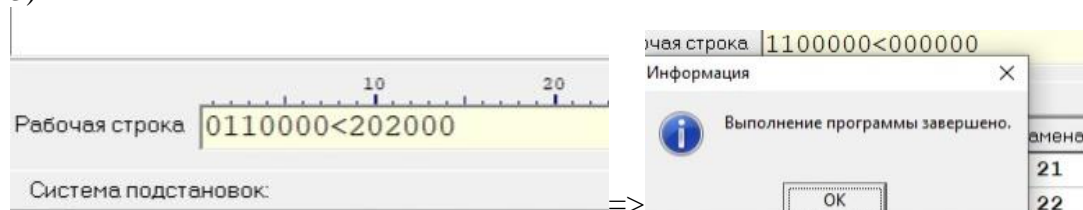
1)



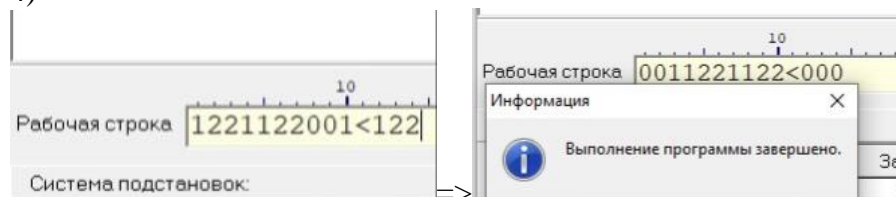
2)



3)



4)



Пункты 1-7 отчета составляются строго до начала лабораторной работы.

Допущен к выполнению работы. Подпись преподавателя \_\_\_\_\_

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

	Образец		Замена		Образец		Замена		Образец		Замена
1	*1	→	1*	30	;2	→	2;	58	0@2	→	\$12
2	*2	→	2*	31	;1	→	1;	59	0@	→	@1
3	*0	→	0*	32	0;<	→	0\$<	60	1\$1	→	@01
4	*<	→	<*	33	1;<	→	0@<	61	1\$	→	\$1
5	-0	→	2-	34	2;<	→	0№<	62	1@	→	@1
6	2#0	→	12-	35	0@<	→	@0<	63	0\$	→	\$0
7	1#0	→	02-	36	0№<	→	№0<	64	1^	→	^1
8	1#	→	01	37	2№<	→	№2<	65	0^	→	^1
9	2#	→	11	38	2\$0	→	№00	66	2\$	→	\$2
10	0#	→	#0	39	1№0	→	@20	67	2@	→	@2
11	0!	→	!0	40	0№1	→	\$21	68	1№	→	№1
12	1!	→	!1	41	1№2	→	@22	69	№00	→	00
13	2!	→	!2	42	2№1	→	№21	70	№01	→	01
14	<!	→	!<	43	2№2	→	№22	71	№10	→	10
15	?1	→	1?	44	0№2	→	\$22	72	№11	→	11
16	?2	→	2?	45	2№0	→	№20	73	№12	→	12
17	?0	→	0?	46	0\$2	→	\$02	74	№02	→	02
18	0?<	→	1\$<	47	2\$1	→	№01	75	№20	→	20
19	1?<	→	1@<	48	0@1	→	\$11	76	№21	→	21
20	2?<	→	1№<	49	1\$0	→	@00	77	№22	→	22
21	^2	→	2^	50	0@0	→	\$10	78	@00	→	00
22	^0	→	0^	51	1№1	→	@21	79	@01	→	01
23	^1	→	1^	52	0№0	→	\$20	80	@20	→	20
24	2^<	→	2№<	53	2\$2	→	№02	81	@21	→	21
25	1^<	→	2@<	54	2@2	→	№12	82	@02	→	02
26	0^<	→	\$2<	55	2@1	→	№11	83	@10	→	10
27	2@<	→	@2<	56	1\$2	→	@02	84	@11	→	11
28	2\$<	→	2\$<	57	2@0	→	№10	85	@12	→	12
29	;0	→	0;	58	0@2	→	\$12	86	@22	→	22
87	\$00	→	00								
88	\$01	→	01								
89	\$10	→	10								
90	\$11	→	11								
91	\$02	→	02								
92	\$12	→	12								
93	\$20	→	20								
94	\$21	→	21								
95	\$22	→	22								
96	\$0	→	0								
97	@1	→	1								
98	№2	→	2								
99	@	→									
100	!2	→	^2								
101	!0	→	;0								
102	!1	→	?1								
103	<#	→	.<								
104	-	→	!								
105	%	→									
106	*	→	#								
107		→	*								

**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

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

### 11.Выводы

Нормальные Алгоритмы Маркова - форма записи алгоритмов обработки символьных строк, которую можно использовать для доказательства разрешимости или неразрешимости задачи. В данной работе я полностью изучил составление алгоритмов на НАМ, что послужит мне неплохой базой для программирования на Си и на других языках!

Подпись студента \_\_\_\_\_