

# Отчет по лабораторной работе № 7 по курсу «Фундаментальная информатика»

Студент группы Алапанова Эльза Халилевна, № по списку 3

Контакты e-mail : alapanowa02@yandex.ru

Работа выполнена: «09» ноября 2020г.

Преподаватель: каф. 806 Найденов Иван Евгеньевич

Отчет сдан «    » \_\_\_\_\_ 20 \_\_\_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_

1. **Тема:** программирование в алгоритмической модели Маркова
2. **Цель работы:** разработать алгоритмическую модель Маркова решения задачи.
2. **Задание (вариант № 28):** составить алгоритм, восстанавливающий целое число в шестнадцатеричной позиционной системе счисления по его обратному коду.
3. **Оборудование** (студенческое)

Процессор Celeron(R) Dual-Core CPU T3300 с ОП \_\_\_\_\_ Мб, НМД \_\_\_\_\_ Мб. Монитор \_\_\_\_\_

#### 4. Программное обеспечение (студенческое):

Операционная система семейства linux, наименование ubuntu версия 14.04 LTS  
интерпретатор команд \_\_\_\_\_ версия \_\_\_\_\_.

Система программирования \_\_\_\_\_ версия \_\_\_\_\_

Редактор текстов \_\_\_\_\_ версия \_\_\_\_\_

Утилиты операционной системы \_\_\_\_\_

Прикладные системы и программы \_\_\_\_\_

Местонахождение и имена файлов программ и данных на домашнем компьютере \_\_\_\_\_

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

Числа в 16-ой системе счисления выглядят так:

0000 0, 0001 1, 0010 2, 0011 3, 0100 4, 0101 5, 0110 6, 0111 7, 1000 8, 1001 9, 1010 A, 1011 B, 1100 C, 1101 D, 1110 E, 1111 F, т.к это обратный код, то по его первому числу нужно смотреть, какое изначально было число: положительным или отрицательным. Если в двоичной системе счисления у первого числа в начале стоит 1-число отрицательное, 0-число положительное.

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

План работы:

1) составляем алгоритм для положительных чисел

2) составляем алгоритм для отрицательных чисел

3) алгоритм для отдельных случаев: когда число состоит только из 0-> в конце должен остаться ноль. Также если есть ведущие нули, то убираем их.

4) рассматриваем случаи, когда первые числа F-> нужно их заменить на нули, стереть и к тому же число должно быть отрицательным.

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

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

\*08->#8| // для случая, если сначала стоял 0, чтобы алгоритм не начал выполнять команды для отрицательного числа

\*09->#9|

\*0A->#A|

\*0B->#B|

\*0C->#C|

\*0D->#D|

\*0E->#E|

\*0F->#F|

\*0->\* // убираем ведущие нули

\*1->#1| //числа, которые в двоичной системе не начинаются с 1

\*2->#2|

\*3->#3|

\*4->#4|

\*5->#5|

\*6->#6|

\*7->#7|

\*8->#-7& //числа, которые в двоичной системе начинаются с 1

\*9->#-6&

\*A->#-5&

\*B->#-4&

\*C->#-3&

\*D->#-2&

\*E->#-1&

\*F->#-&

#&0->#&

#&F->#&

#-&0->#-F& //отдельный случай, когда первым стоит F

#-&1->#-E&

#-&2->#-D&

#-&3->#-C&

#-&4->#-B&

#-&5->#-A&

#-&6->#-9&

#-&7->#-8&

#-&8->#-7&

#-&9->#-6&

#-&A->#-5&

#-&B->#-4&

#-&C->#-3&

#-&D->#-2&

#-&E->#-1&

#-&F->#-0&

#-0->#-

#&->0\*#

&0->F& // восстановление из обратного кода при отрицательном числе

&1->E&

&2->D&

&3->C&

&4->B&

&5->A&

&6->9&

&7->8&

&8->7&

&9->6&

&A->5&

&B->4&

&C->3&

&D->2&

&E->1&

&F->0&

F&->F //конец восстановления при отрицательном числе

E&->E

D&->D

C&->C

B&->B

A&->A

9&->9

8&->8

7&->7  
6&->6  
5&->5  
4&->4  
3&->3  
2&->2  
1&->1  
0&->0

|0->0| //восстановление при положительном числе

|1->1|  
|2->2|  
|3->3|  
|4->4|  
|5->5|  
|6->6|  
|7->7|  
|8->8|  
|9->9|  
|A->A|  
|B->B|  
|C->C|  
|D->D|  
|E->E|  
|F->F|  
F|->F  
E|->E  
D|->D  
C|->C  
B|->B  
A|->A  
9|->9  
8|->8  
7|->7  
6|->6  
5|->5  
4|->4  
3|->3  
2|->2  
1|->1  
0|->0

\*#->. //конец алгоритма  
-&->0 //отдельный случай  
\*->0\*#  
->\*

**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						
2						

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

**11. Выводы :** работа далась тяжело. Изначально неправильно поняла суть обратного кода, поэтому ничего не получалось. Затем, путем разбора ошибок, удалось прийти к чему-то хорошему. Данная работа и вправду заставляет включить хоть какое-то мышление. В общем можно сказать, что было интересно находить свои ошибки и что-то понимать, но из-за своих глупых ошибок было потрачено много времени. Часто появлялась ошибка именно с ведущими нулями, либо ноль совсем стирался, либо один оставался в начале числа.

Недочёты при выполнении задания могут быть устранены следующим образом: \_\_\_\_\_

\_\_\_\_\_ Подпись студента \_\_\_\_\_