

Отчет по лабораторной работе № 12 по курсу “Фундаментальная информатика”

Студент группы М80-109Б-22 Серякова Александра Андреевна, № 17

Работа выполнена: «02» октября 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

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

Подпись преподавателя _____

1. **Тема:** Техника работы с целыми числами. Системы счисления.
2. **Цель работы:** Составить программу на языке Си в целом типе данных, которая для любых допустимых и корректно записанных чисел этого типа в десятичном изображении, поступающих на стандартный ввод программы, выполняет указанное вариантом действие над их значениями.

2. Задание (вариант № 28):

28. Получить двоично-кодированное десятичное представление числа

3. Оборудование (студента):

Процессор AMD Ryzen 5 5500U with Radeon Graphics @ 2.100GHz с ОП 9812 Мб, SSD 512 Гб.
Монитор 1920x1080

4. Программное обеспечение (студента):

Операционная система семейства: linux, наименование: Arch x86_64
интерпретатор команд: bash версия 5.1.16
Система программирования -- версия --, редактор текстов neo vim версия 0.7.2
Утилиты операционной системы mkdir, cd, touch, ls, echo, cat, find, grep, rm, chmod, bash, pwd
Прикладные системы и программы –
Местонахождение и имена файлов программ и данных на домашнем компьютере
/home/taida/Programming/MAI_labs/lab5

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

Заношу цифры введенного мной числа в массив (путем нахождения остатков этого числа при делении его на 10 до тех пор, пока число >0) Но цифры в массивы оказываются в обратном порядке, во избежание это создаю новый массив, в котором я восстановлю последовательность цифр. Далее Прохожусь по массиву и каждый его элемент(цифру введенного мной числа) перевожу в двоичную систему счисления. Всё!!!!

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

Проводилось unit-тестирование, исходный код тестов приложен в пункте №8.

Тесты:

Входные данные	результат
456	010001010110

27891	00100111100010010001
111111111	000100010001000100010001000100010001
123456789	000100100011010001010110011110001001

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

```
1  #include <stdio.h>
2  #include <assert.h>
3
4
5  int dv(int a){
6      if (a == 0){
7          printf("0000");
8          return 0000;
9      }
10     if (a == 1){
11         printf("0001");
12         return 0001;
13     }
14     if (a == 2){
15         printf("0010");
16         return 0010;
17     }
18     if (a == 3){
19         printf("0011");
20         return 0011;
21     }
22     if (a == 4){
23         printf("0100");
24         return 0100;
25     }
26     if (a == 5){
27         printf("0101");
28         return 0101;
29     }
30     if (a == 6){
31         printf("0110");
32         return 0110;
33     }
34     if (a == 7){
35         printf("0111");
36         return 0111;
37     }
38     if (a == 8){
39         printf("1000");
40         return 1000;
41     }
42     if (a == 9){
43         printf("1001");
44         return 1001;
45     }
46 }
47
48 void unit_test(){
49     assert(dv(2)==0010);
50     assert(dv(1)==0001);
51     assert(dv(7)==0111);
52 }
53
54 int main()
55 {
56     int num,k,i,cnt;// num и k- наше число, cnt-количество цифр в числе
57     int *arr;
58     int *tarr;
59
60     scanf("%d",&num);
61
62     k = num;
63     cnt = 0;
64
65     while(k != 0) //цикл чтобы найти количество цифр в числе
66     {
67         k = k / 10;
68         cnt++; //увеличим количество цифр в числе
69     }
70
71     arr = (int*)malloc(cnt * sizeof(int));
72     tarr = (int*)malloc(cnt * sizeof(int));
73
74
75     for(i=0;i<cnt;i++)
76     {
77         arr[i] = num%10; //берем последние цифры с числа
78
79         num = num / 10; //забираем эти цифры с числа
80     }
81
82     for (i = 0; i<cnt; ++i){
83         tarr[i] = arr[cnt-1-i]; // в новый массив заносим старый arr, но уже в перевернутом порядке
84     }
85
86     /*for (i = 0;i<cnt; ++i){
87         printf("%d", tarr[i]); // выводим на экран массив tarr
88     }
89     printf("\n");*/
90
91     for (i = 0; i<cnt; ++i){
92         dv(tarr[i]);
93     }
94
95     return 0;
96 }
97 }
```

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	Время	Событие	Действие по исправлению	Примечание
Дом	12.12.22	4:50	Даже не знаю что сказать.... 5 часов, а я пишу отчет.	Да-да нужно скорее принимать действия по исправлению этого недоразумения и идти спать	Не судите строго, реально спать пора...

10. Замечания автора по существу работы

11. Выводы

Крутая-супер лаба, все понравилось, все супер и замечательно, достаточно легкая. 5 часов, спать пора – вроде нормальный вывод для такого времени... 😊

Недочёты при выполнении задания могут быть устранены следующим образом: --

Подпись студента _____