

Отчет по лабораторной работе №12 по курсу _____ Фундаментальная информатика _____

Студент группы _____ М8О-102Б-21 _____, № по
списку _9_ - Кажекин Денис Андреевич

Контакты www, e-mail, icq,
skype _____ deniskazhekin@mail.ru _____

Работа выполнена:

Преподаватель: _____ Никулин Сергей Петрович _____
каф. 806 _____

Входной контроль знаний с оценкой

Отчет сдан « _____ » _____ 201 ____ 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 любимых допустимых и корректно записанных чисел этого типа в десятичном изображении, поступающих на стандартный ввод программы, выполняет указанное вариантом действие над их значениями
3. **Задание:** после каждых двух цифр вставить их сумму, если она также является цифрой (Вариант 26)
4. **Оборудование:**

Оборудование ПЭВМ студента, если использовалось:

Процессор: Apple M1, с ОП 8192 Мб, НМД 262144 Мб. Монитор: Retina 13,3; IPS 2560 x 1600.

5. Программное обеспечение ЭВМ студента, если использовалось

Операционная система семейства Linux, наименование Ubuntu, версия _____ 20.04.3 LTS _____ интерпретатор команд `_bash_` версия _____ 5.0.17(1) _____

Редактор текстов GNU emacs, версия 27.2

Прикладные системы и программы: Компилятор языка C – gcc.

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

1. Числа, состоящие из одной цифры, рассматриваем отдельно, сразу выводя их в ответ
2. В остальных же случаях для начала найдем длину нашего числа и в числах нечетной длины сразу же удалим левую цифру, записав её в ответ, потому что она все равно не будет ни на что влиять. Таким образом, теперь у нас все числа будут одного вида (четной длины)
3. За одну итерацию цикла “while” будем считывать две левые цифры и записывать в ответ новую цифру – сумму, а затем эти две цифры, при этом проверяя сумму на то, чтобы она была меньше 9, а также следим за тем, чтобы не было переполнения переменной “int”. В конце итерации не забываем “отрезать” использованные две левые цифры.
7. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

Тесты:

1. 123 >> 1523
2. 1010 >> 110110
3. -100 >> -1000
4. 0 >> 0
5. 21412412 >> Warning! Check your input data
6. 13579 >> 183579
7. 7777 >> 7777
8. 19999 >> 19999
9. 292292 >> 2942292
10. 5445 >> 954945

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).

denis@ubuntu:~\$ cat lr12.c

```
#include "stdio.h"
#include "math.h"

int main(){
    int input, left, right, final=0, n, data, to_n, final_n, max_int = 2147483647, flagz = 1, chislo, kn;
    char flag = 'y';

    while (flag != 'n') {
        final = 0;
        n = 0;
        to_n = 0;
        flagz = 1;
        printf("Enter input data: ");
        scanf("%i", &input);
        data = input;
        if (input >= -9 && input <= 9) {
            printf("%d\n", input);
        }
        else {
            while (data != 0) {
                data = data / 10;
                n += 1;
            }
            if (n % 2 == 1) {
                final = input / pow(10, n - 1);
                input = fmod(input, pow(10, n - 1));
                final_n = (n - 1) / 2;
                kn = n - 1;
            }
            else {
                final_n = n / 2;
                kn = n;
            }
            while (to_n < final_n) {
                if (input > 99) {
                    chislo = input / pow(100, (kn / 2) - 1);
                }
                if (input <= 99) {
                    chislo = fmod(input, 100);
                }
                left = chislo / 10;
                right = chislo % 10;
                if (left + right <= 9) {
                    if (max_int / 10 >= final) {
                        final = (final * 10) + (left + right);
                    } else {
                        flagz = 0;
                        break;
                    }
                }
                if (max_int / 10 >= final) {
                    final = (final * 10) + left;
                } else {
                    flagz = 0;
                    break;
                }
                if (max_int / 10 >= final) {
                    final = (final * 10) + right;
                } else {

```

```

        flagz = 0;
        break;
    }
}
else {
    if (max_int / 10 >= final) {
        final = (final * 10) + left;
    } else {
        flagz = 0;
        break;
    }
    if (max_int / 10 >= final) {
        final = (final * 10) + right;
    } else {
        flagz = 0;
        break;
    }
}
input = fmod(input, pow(100, (kn / 2) - 1));
to_n += 1;
kn -= 2;
}

if (flagz == 1) {
    printf("%d\n", final);
} else {
    printf("Warning! Check your input data\n");
}
}
printf("Do you want to continue (y/n) ? : ");
scanf("%s", &flag);
}
return 0;
}

```

denis@ubuntu:~\$ emacs lr12.c

denis@ubuntu:~\$ gcc lr12.c -lm

denis@ubuntu:~\$./a.out

Enter input data: 123

1523

Do you want to continue (y/n) ? : y

Enter input data: 1010

1101110

Do you want to continue (y/n) ? : y

Enter input data: -100

-1000

Do you want to continue (y/n) ? : y

Enter input data: 0

0

Do you want to continue (y/n) ? : y

Enter input data: 21412412

Warning! Check your input data

Do you want to continue (y/n) ? : y

Enter input data: 13579

183579

Do you want to continue (y/n) ? : y

Enter input data: 7777

7777

Do you want to continue (y/n) ? : y

```

Enter input data: 19999
19999
Do you want to continue (y/n) ? : y
Enter input data: 292292
2942292
Do you want to continue (y/n) ? : y
Enter input data: 5445
954945
Do you want to continue (y/n) ? : n
denis@ubuntu:~$

```

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.

	Ла б. или до м.	Д а т а	Вре мя	Событие	Действие по исправлению	Примечание

10. Замечания автора по существу работы

11. Выводы я научился составлять программы, написанные на Си, для работы с целыми числами

Подпись студента _____