

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

Студент группы М80-109Б-22 Любарский Иван Владимирович, № по списку 8

Контакты ivanred289@gmail.com, @rMeDGranD

Работа выполнена: «7» декабря 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

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

Подпись преподавателя \_\_\_\_\_

1. **Тема:** Техника работы с целыми числами
2. **Цель работы:** Составить программу на языке Си для анализа и обработки целочисленных значений.
3. **Задание:** Инкрементировать все четные цифры. (Вариант 24)

4. **Оборудование:**

Процессор *Intel Core i5-4210U @ 4x 1.7GH* с ОП *15873* Мб, НМД *512* Гб. Монитор *1600x900*

5. **Программное обеспечение:**

Операционная система семейства: *linux*, наименование: *ubuntu*, версия *22.04 LTS*

интерпретатор команд: *bash* версия *4.4.19*.

Система программирования -- версия --, редактор текстов *emacs* версия *25.2.2*

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)

Для решения поставленной задачи следует создать алгоритм проверки на четность каждой цифры числа. Это можно получить целочисленным делением начального числа на 10 и считывания его остатка на 10.

Если цифры оказывается четной, тогда добавляем ее в новое число, увеличив при это на единицу. Иначе просто добавляем ее в новое число.

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

## План работы программы:

1. Считывание числа
2. Проверка цифры числа на четность
3. Инкремент/не инкремент цифры
4. Создание нового числа
5. Вывод нового числа

Входные данные	Выходные данные	Описание тестируемого случая
0	1	Преобразование 0
122334	133335	Преобразование целого числа произвольной длины
-122334	-133335	Преобразование целого числа в отрицательном представлении
0000000000	1	Преобразование незначащих нулей

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

```
#include <stdio.h>
#include <assert.h>

long long IncrementChet(long long number){
int negative = 0;
if (number==0) { return 1; }
if (number < 0) { number *= -1; negative = 1; }
long long result = 0;
long long R = 1;
while(number>0){
if ((number%10)%2 == 0){ result = result + ((number%10)+1)*R; }
else { result += (number%10)*R; }
R *= 10;
number/=10;
}
if (negative) { return -result; }
else { return result; }
}

void test(){ assert(IncrementChet(10)==11); }
int main(){
test();
long long number;
while(scanf("%lli", &number)){ printf("%lli%c", IncrementChet(number), ' '); }
return 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

**10. Замечания автора** по существу работы

**11. Выводы**

Работа нацелена на расширение понимания целочисленного типа данных студентом. Это включает целочисленное деление, остаток и т. д. При изучении языка это практически первое, что нужно изучить, так как любая численная работа будет напрямую связана с ними.

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

Подпись студента \_\_\_\_\_