

# Отчет по лабораторной работе № 9 по курсу “Фундаментальная информатика”

Студент группы М8О-109Б-22 Нгуен Ньы Хоанг Ань, № по списку 12

Контакты vk, telegram @tng00

Работа выполнена: «07» ноября 2022 г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

Отчет сдан « » 2022 г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_

1. **Тема:** Программа на языке C.

2. **Цель работы:** Составление и отладка простейшей программы на языке C итеративного характера с целочисленными рекуррентными соотношениями.

3. **Задание:** вариант №3.

4. **Оборудование** (студента):

Процессор *Intel(R) Core(TM) i3-7100U CPU @ 2.40GHz* с ОП 5,88 Гб, НМД 1024 Гб. Монитор 1920x1080

5. **Программное обеспечение** (студента):

Операционная система семейства: *linux*, наименование: *ubuntu*, версия *22.04.1 LTS*

Интерпретатор команд: *bash* версия *5.1.16*.

Система программирования -- версия --, редактор текстов *emacs* версия 28.2

Утилиты операционной системы --

Прикладные системы и программы --

Местонахождение и имена файлов программ и данных на домашнем компьютере /home/tng00

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

Пишем функции *sign*, *min*, *max* и *mod*. Запускаем цикл, который:

- Изменяет *i*, *j*, *l* по заданным правилам.
- Проверяет, нет ли точки в области – кольцо, ограниченное двумя окружностями с центром в точке (10, 10), радиус большей – 10, меньшей 5. Если результат положительный, то выводим шаг, на котором это случилось, заканчиваем программу.
- Если счётчик шагов становится равным 50 выводит координаты точки, параметр движения и заканчивает работу.

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

```
#include <stdio>

int min (int x, int y) {
    return (x < y) ? x : y;
}

int max (int x, int y) {
    return -min(-x, -y);
}

int sign (int x) {
    if (x < 0) {
        return -1;
    }
    if (x > 0) {
        return 1;
    }
    return 0;
}

int abs (int x) {
```

```

    return (x < 0) ? -x : x;
}

int mod (int x, int m) {
    int r = x - (x / m) * m;
    return (r >= 0) ? r : r + abs(m);
}

int main() {
    const int i0 = 1,
              j0 = -30,
              l0 = 1;
    int ip = i0,
        jp = j0,
        lp = l0, ik, jk, lk;
    for (int i = 1; i <= 50; ++i) {
        int k = i - 1;
        ik = max(min(ip + jp - lp - k, ip - jp + lp - k), min(k + ip - jp - lp, k -
ip - jp + lp));
        jk = jp + mod(lp * sign(jp), 20) + mod(k * sign(ip), 10);
        lk = abs(ip - jp + lp - k) * sign(ip) * sign(jp);
        ip = ik,
        jp = jk,
        lp = lk;
        int res = (ik - 10) * (ik - 10) + (jk - 10) * (jk - 10);
        if (res >= 25 && res <= 100) {
            printf("Success at step #%d\n"
                  "Coordinates: (%d, %d)\n"
                  "Motion parameter: %d", i, ik, jk, lk);
            break;
        }
        if (i == 50) {
            printf("Fail\n"
                  "Coordinates: (%d, %d)\n"
                  "Motion parameter: %d", ik, jk, lk);
        }
    }
}

```

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

C:\Users\tngue\CLionProjects\untitled\cmake-build-debug\untitled.exe  
Success at step #2  
Coordinates: (8, 2)  
Motion parameter: -8  
Process finished with exit code 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**

В ходе выполнения лабораторной работы я научился основам программирования на языке C.

Подпись студента

