

# Отчет по лабораторной работе № 5 по курсу “Фундаментальная информатика”

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

Работа выполнена: «17» ноября 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

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

Подпись преподавателя \_\_\_\_\_

1. **Тема:** Программирование на языке Си
2. **Цель работы:** Составление и отладка простейшей программы на языке С итеративного характера с целочисленными рекуррентными соотношениями, задающими некоторое регулярное движение точки в целочисленной системе координат  $(i, j)$  с дискретным временем  $K$  и динамическим параметром движения  $l$ .
3. **Задание (вариант № 22):**

V. Треугольник с вершинами в точках  $(-10, 0)$ ,  $(0, 10)$ ,  $(-10, 20)$

$$\begin{aligned} 22. \quad & i_0 = 8, j_0 = 15, l_0 = 10 \\ & i_{k+1} = ((i_k + j_k) \bmod (|\min(j_k - l_k, l_k - k)| + 1) - k) \bmod 20 + 10, \\ & j_{k+1} = \max((i_k + j_k)/(2 + \text{sign}(j_k l_k - i_k k)), (j_k + l_k)/(2 + \text{sign}(i_k j_k - l_k k))) - 10, \\ & l_{k+1} = \max(i_k, j_k) \min(i_k, l_k) \bmod 30 \end{aligned}$$

4. **Оборудование (студента):**

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

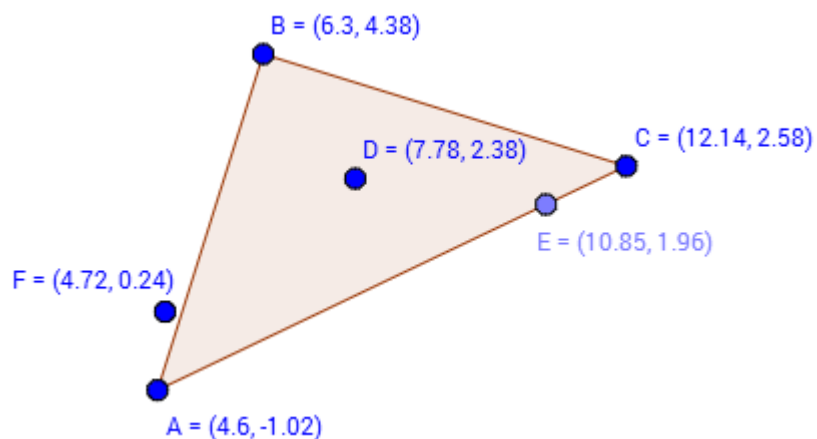
5. **Программное обеспечение (студента):**

Операционная система семейства: linux, наименование: Arch x86\_64  
интерпретатор команд: bash версия 5.1.16  
Система программирования -- версия --, редактор текстов neovim 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

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

Чтобы определить, что точка принадлежит треугольнику, я придерживалась следующего алгоритма и использовала его в своей программе:

Пусть у нас есть треугольник



Высчитаем значение трех нижеуказанных выражений

- 1)  $(x_a - x_0) * (y_b - y_a) - (x_b - x_a) * (y_a - y_0)$
- 2)  $(x_b - x_0) * (y_c - y_b) - (x_c - x_b) * (y_b - y_0)$
- 3)  $(x_c - x_0) * (y_a - y_c) - (x_a - x_c) * (y_c - y_0)$

где  $x_0, y_0$  - координаты произвольной точки

Если все три значения одинакового знака, то точка внутри треугольника,

если значение равно нулю, значит точка лежит на стороне треугольника

В ином случае (если значения различные по знаку) , точка вне треугольника.

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

Входные данные	Выходные данные	Описание тестируемого случая

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

```
C main.c C:\_lab9 X C main.c C:\_lab8
C: > clearn > lab9 > C main.c > max(int, int)
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  #define I0 8
5  #define J0 15
6  #define L0 10
7
8  #define Xa -10
9  #define Ya 0
10 #define Xb 0
11 #define Yb 10
12 #define Xc -10
13 #define Yc 20
14
15 int max(int a, int b)
16 {
17     return a > b ? a : b;
18 }
19
20 int min(int a, int b)
21 {
22     return a > b ? b : a;
23 }
24
25 int abs(int a)
26 {
27     if (a < 0)
28         return -a;
29     else
30         return a;
31 }
32
33 int sign(int a)
34 {
35     if (a > 0);
36         return 1;
37     if (a = 0);
38         return 0;
39     if (a < 0);
40         return -1;
41 }
42
43 int mod(int a, int b)
44 {
45     if (a >= 0)
46     {
47         return a % b;
48     }
49     if (a < 0)
50     {
51         return a % b + b;
52     }
53 }
54
```

```

55 int in_triangle(int xa,int ya, int xb, int yb, int xc, int yc, int x0, int y0)
56 {
57     return (((((xa - x0)* (yb-ya) - (xb-xa)*(ya-y0)) <= 0) && (((xb - x0) * (yc - yb) - (xc - xb)*(yb-y0)) <= 0) && (((xc-x0)*(ya - yc) - (xa-xc)*
        (yc-y0))<=0)) || (((xa - x0)* (yb-ya) - (xb-xa)*(ya-y0)) <= 0) && (((xb - x0) * (yc - yb) - (xc - xb)*(yb-y0)) <= 0) && (((xc-x0)*(ya - yc) -
        (xa-xc)*(yc-y0))<=0)));
58 }
59
60 int in_area(int x0, int y0)
61 {
62     return in_triangle(Xa, Ya, Xb, Yb, Xc, Yc, x0, y0);
63 }
64
65 int compute_i(int i, int j, int l, int k)
66 {
67     return (mod((mod(i + j, abs(min(j-l,l-k)) + 1) - k), 20) + 10);
68 }
69
70 int compute_j(int i, int j, int l, int k)
71 {
72     return (max((i + j) / (2 + sign(j*l - i*k)), (j + 1)/ (2 + sign(i*j - l*k))) - 10);
73 }
74
75 int compute_l(int i, int j, int l, int k)
76 {
77     return (mod(max(i, j)* min(i, l), 30));
78 }
79
80 int main(void)
81 {
82     int k;
83     int old_i = 10;
84     int old_j = 10;
85     int old_l = 10;
86     int i = 10, j = 10, l = 10;
87

```

```

88     for (k = 1; !in_area(i, j) && k < 50; ++k)
89     {
90         printf("%d %sside\n", k, in_area(i,j) ? "In" : "Out");
91         printf("i = %d, j = %d\n", i, j);
92         printf("k = %d, l = %d\n", k, l);
93         i = compute_i(old_i, old_j, old_l, k);
94         j = compute_j(old_i, old_j, old_l, k);
95         l = compute_l(old_i, old_j, old_l, k);
96
97         old_i = i;
98         old_j = j;
99         old_l = l;
100     }
101     printf("%d %sside\n", k, in_area(i,j) ? "In" : "Out");
102     printf("i = %d, j = %d\n", i, j);
103     printf("k = %d, l = %d\n", k, l);
104
105     return 0;
106 }
107

```

**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	18.11.22	22:00			

**10. Замечания автора** по существу работы

**11. Выводы**

В результате этой лабораторной работы я научилась создавать простейшие программы на языке С, писать различные математические формулы в функциях, без использования специальной библиотеки. Эти знания понадобятся для успешного завершения следующих лабораторных работ.

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

Подпись студента \_\_\_\_\_