Отчёт по лабораторной работе № 9 по курсу «Архитектура компьютера и информационных систем»

Студент группы М8О-112Б-22

Епифанов Евгений Валерьевич, № по списку: 9

# E-mail: [epi\_abitur](mailto:epi_abitur)@mail.ru

# Телефон: 8(916)783-95-69

# Работа выполнена: 18 сентября 2022 г.

Преподаватель: Никулин Сергей Петрович, каф.806

**1. Тема:** системы программирования 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 системы координат (i, j) с дискретным временем k и динамическим параметром движения l.

**3. Задание:** составить и отладить программу на Си в терминальном классе для решения полученной задачи.

Изображение выглядит как текст

Автоматически созданное описание

(Необходимо рассчитать, попадает ли точка, заданная координатами i и j, в область в виде пересечения двух окружностей за 50 шагов, где на каждом шаге значения i и j изменяются.) Вариант 13: “Лунка, являющаяся пересечением двух кругов радиуса 10, центр первого круга – в точке (-10, -10), центр второго – в точке (-20, -20).”.

**4. Оборудование (ПЭВМ студента):**

Ноутбук с процессором Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz, 1992 МГц, ядер: 4, логических процессоров: 8, ОП 8 ГБ. SSD 256 ГБ.

**5. Программное обеспечение ЭВМ студента:**

* Операционная система семейства UNIX, наименование: Linux Ubuntu (22.04) 5.15.0-48-generic #54-Ubuntu SMP Fri Aug 26 13:26:29 UTC 2022 x86\_64 x86\_64 x86\_64 GNU/Linux.
* Интерпретатор команд: GNU bash, version 5.1.16(1)-release (x86\_64-pc-linux-gnu).
* Компилятор gcc Linux.
* Редактор текстов: GNU Emacs 28.2
* Утилиты и прочие программы.

**6. Идея, метод, алгоритм решения задачи:**

1. Взаимодействие с программой.
   1. emacs \_\_\_.c – запуск emacs с последующим вводом кода.
   2. cc \_\_.c -lm – компиляции программы с использованием библиотек.
   3. ./a.out – тестирование программы.
2. Начальные условия:

Изображение выглядит как текст

Автоматически созданное описание

1. Программа:

/\*

Лабораторная №9, программа - отслеживание попадания точки в заданную область

студент группы М80-112Б-22, Епифанов Е.В.

\*/

#include <stdio.h>

#include <math.h>

int mod(int n, int m) // Функция mod

{

return(((n%m)+m)%m);

}

int main()

{

long long R = 10, a1 = -10, b1 = -10, a2 = -20, b2 = -20, i = 13, j = -9, l = -4, di, dj, dl;

int k;

for (k = 0; k <= 50; ++k)

{

di = i;

dj = j;

dl = l;

if ( ((((a1 - i)\*(a1 - i)) + ((b1 - j)\*(b1 - j))) < R\*R) && ((((a2 - i)\*(a2 - i)) + ((b2 - j)\*(b2 - j))) < R\*R))

{

printf("%s%s%lli%s%lli%s%lli%s%i\n", "Попадание!", " x=", i, " y=", j, " l=",l , " k=", k);

return 1;

}

i = (((mod((i + j), 30)) / (mod((fabs(l)), 5) + 1)) + ((mod((i + l), 30)) / (mod((fabs(j)), 5) + 1)));

j = ((mod((fmax(k\*i, (k + 1)\*j)), 25)) - ((fabs(j - l)) / 10));

l = (((fabs(j - l)) / 10) + (fmin((mod((i + l), 20)), (mod(j\*k, 20)))) - 10);

}

printf("%s%s%lli%s%lli%s%lli%s%i\n", "Промах!", " x=", di, " y=", dj, " l=", dl, " k=", k - 1);

return 0;

}

Данная программа имеет несколько констант, которые обозначают радиус, точки центров окружностей, координаты нашей точки и параметр. Также были добавлены переменные d\_ для удобного вывода данных в случае, если точка не попала в заданную область. В цикле проверяется, попала ли точка в заданную область, после чего программа находит следующие значения координат нашей точки и параметра. В случае, если точка попадает в нужную область, программа завершается и выводит соответствующее сообщение, аналогично в случае, если точка за 50 шагов не попала в нужную область.

Ответ выводится в формате: “Ответ x=i y=j l=l k=k”, где i и j это координаты необходимой точки.

В данной программе можно использовать любой из трех циклов (for, while, do while), так как в любом случае мы сможем написать корректное условие для выхода из цикла – переменная от 0 до 50. Я использовал цикл for, так как в нём автоматически увеличивается номер k, который необходимо использовать в выводе.

**7. Сценарий выполнения работы:**

1. Проанализировать полученную задачу и разработать для её решения корректно функционирующую программу.
2. Написать программу в текстовом редакторе Emacs.
3. Скомпилировать программу и проверить её работоспособность.

**8. Распечатка протокола:**

evgeniy2@evgeniy2:~$ emacs laba9.c

Изображение выглядит как текст

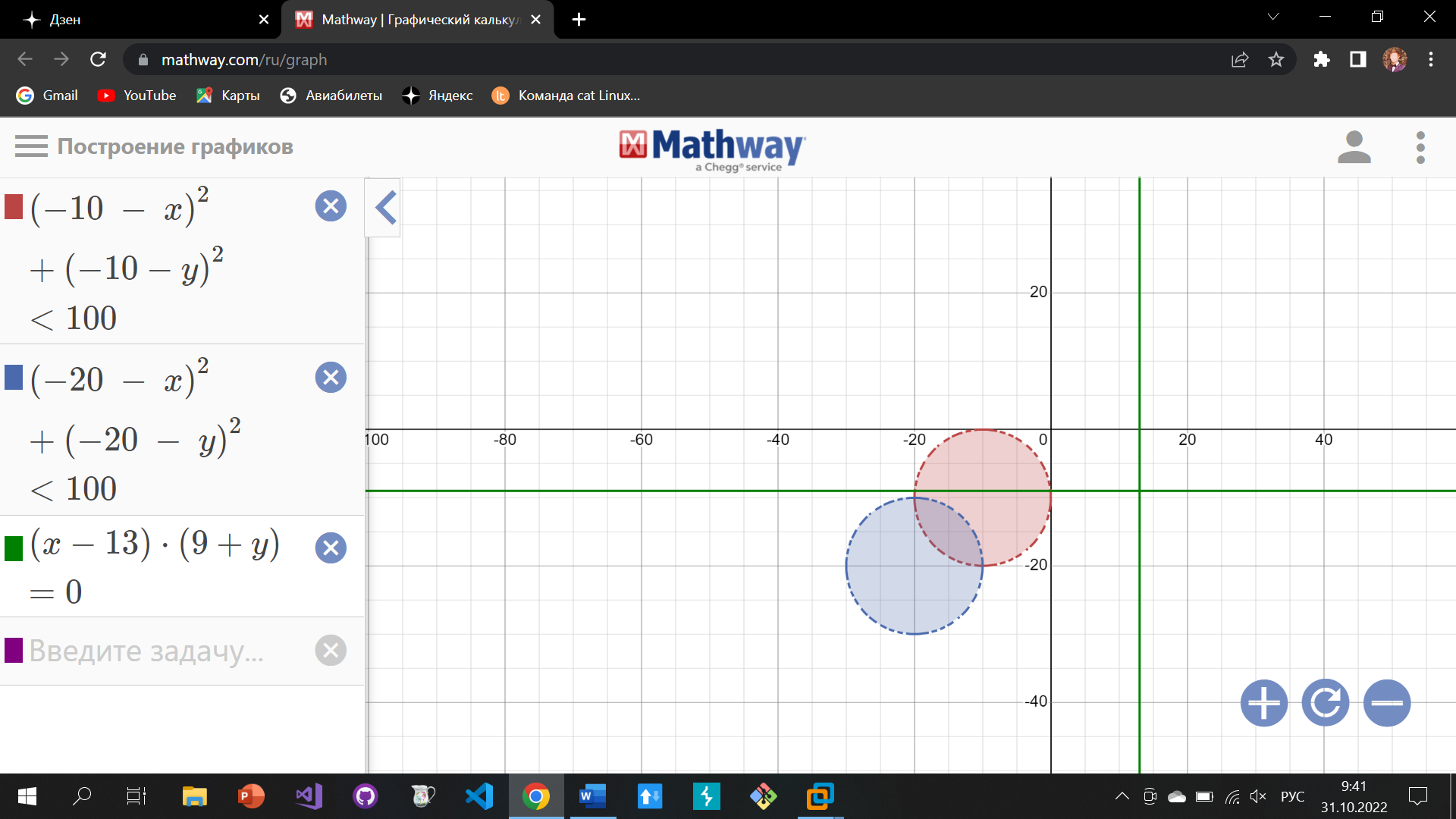
Автоматически созданное описание

evgeniy2@evgeniy2:~$ cc laba9.c -lm

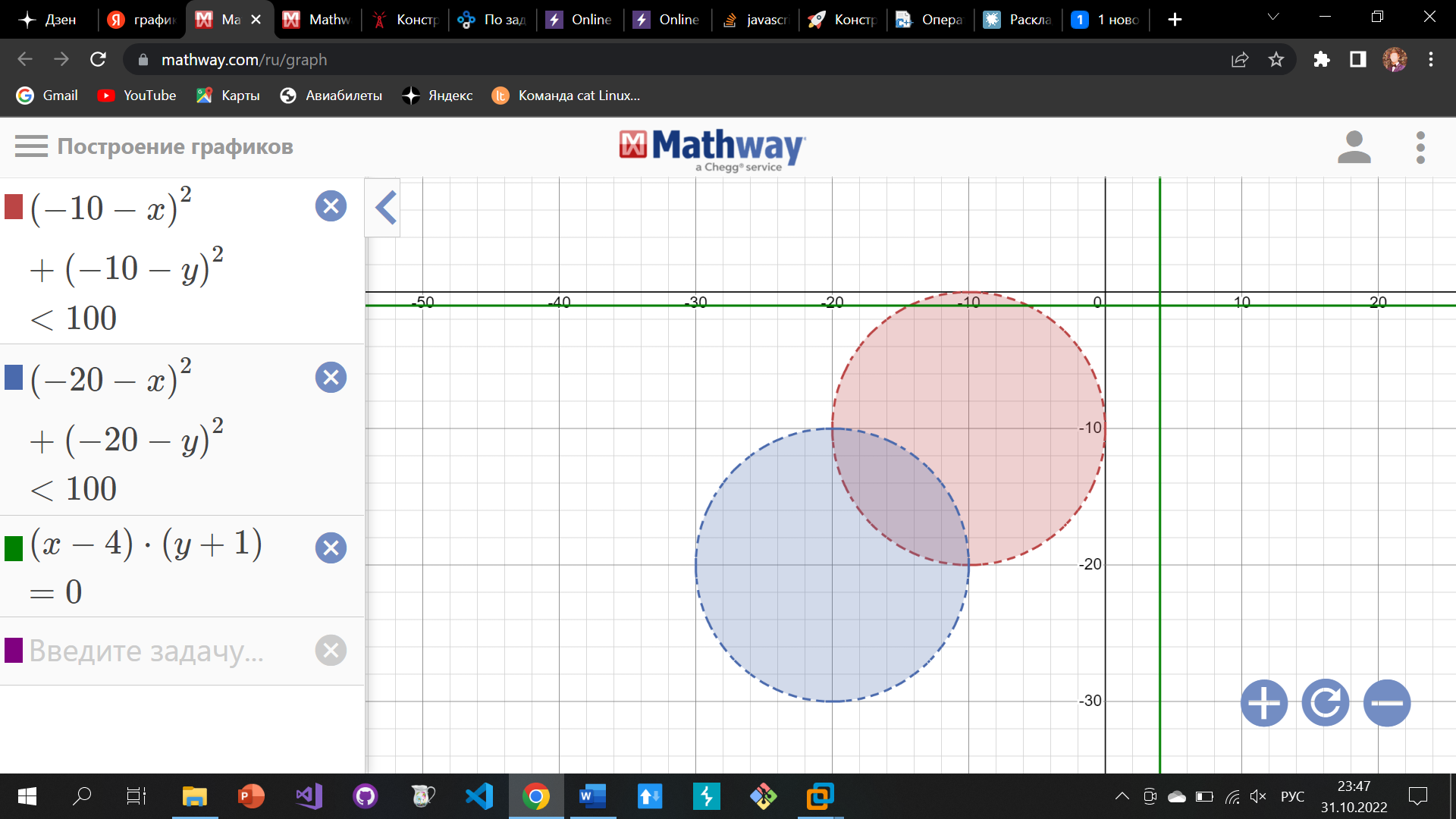
evgeniy2@evgeniy2:~$ ./a.out

Промах! x=4 y=-1 l=1 k=50

Начальное положение точки:



Конечное положение точки:



Точка пересечения двух прямых имеет координаты (4, -1), которая является последним полученным ответом программы. Как видно на графике, эта точка не попадает на пересечение двух окружностей.

Проверка точности программы (вывод данных на каждой итерации):

Обращаем внимание на первые 2 числа в каждой строке – это i и j полученной точки.

evgeniy2@evgeniy2:~$ cc laba9.c -lm

evgeniy2@evgeniy2:~$ ./a.out

x=13 y=-9 l=-4 k=0

x=1 y=0 l=-9 k=1

x=22 y=21 l=-6 k=2

x=14 y=10 l=-8 k=3

x=12 y=13 l=-3 k=4

x=8 y=13 l=-3 k=5

x=6 y=1 l=-6 k=6

x=3 y=17 l=-5 k=7

x=29 y=0 l=-9 k=8

x=25 y=0 l=-9 k=9

x=21 y=13 l=4 k=10

x=6 y=17 l=1 k=11

x=13 y=2 l=-7 k=12

x=7 y=8 l=-8 k=13

x=10 y=3 l=-6 k=14

x=7 y=22 l=-6 k=15

x=14 y=0 l=-9 k=16

x=7 y=11 l=8 k=17

x=11 y=22 l=5 k=18

x=8 y=16 l=0 k=19

x=28 y=5 l=-1 k=20

x=28 y=9 l=-9 k=21

x=4 y=21 l=-6 k=22

x=26 y=19 l=-7 k=23

x=8 y=3 l=-8 k=24

x=2 y=-1 l=4 k=25

x=3 y=0 l=-9 k=26

x=24 y=23 l=8 k=27

x=4 y=17 l=2 k=28

x=9 y=16 l=0 k=29

x=29 y=14 l=-2 k=30

x=9 y=7 l=-2 k=31

x=7 y=23 l=-2 k=32

x=1 y=6 l=2 k=33

x=3 y=3 l=-4 k=34

x=8 y=21 l=-3 k=35

x=9 y=3 l=-4 k=36

x=3 y=10 l=-8 k=37

x=28 y=9 l=-8 k=38

x=5 y=0 l=-9 k=39

x=27 y=2 l=9 k=40

x=7 y=4 l=-9 k=41

x=7 y=10 l=1 k=42

x=16 y=21 l=-6 k=43

x=8 y=21 l=-5 k=44

x=30 y=17 l=-2 k=45

x=14 y=5 l=-4 k=46

x=13 y=22 l=1 k=47

x=6 y=3 l=-8 k=48

x=9 y=5 l=-8 k=49

x=4 y=-1 l=1 k=50

Промах! x=4 y=-1 l=1 k=50

Можно заметить, что x в каждой строке больше 0. Таким образом, ни одна точка не попадет в пересечение окружностей, так как обе окружности полностью находятся в 3 четверти графика.

9. Дневник отладки:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. или дом. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  |  |  |  |  |  |  |

10. Замечания по существу работы: -

11. Выводы: Я научился решать задачи, связанные с графиками, на языке программирования Си. Я усовершенствовал свои навыки программирования на Си в Linux.

Недочёты при выполнении задания могут быть устранены следующим образом: -