

Отчет по лабораторной работе № 8,9 по курсу «Фундаментальная информатика»

Студент группы М8О-105Б-21 Минеева Светлана Алексеевна, № по списку 14

Контакты e-mail: svetlana.mineewa2003@yandex.ru

Работа выполнена: «3» ноября 2021 г.

Преподаватель: Вячеслав Константинович Титов каф. 805

Отчет сдан «3» ноября 2021 г., итоговая оценка _____

Подпись преподавателя _____

1. Тема: Системы программирования на языке Си

2. **Цель работы:** Лабораторная работа №8: Изучение конкретной системы программирования на Си и получение навыков подготовки текстов и отладки программ. Лабораторная работа №9: Составление и отладка простейшей программы на языке Си 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 №5): Лабораторная работа №8:

- 1) Изучить и освоить возможности лабораторной СП по содержащейся в практикуме документации и другим источникам, в том числе основные этапы процесса компиляции и подготовки программ к выполнению;
- 2) Составить и отладить простейшую программу на Си в терминальном классе;
- 3) Изучить различные системы программирования на Си на платформе GNU.

Лабораторная работа №9:

Необходимо определить попадание движущейся по двумерной целочисленной сетке точки в специальную геометрическую область не более чем за 50 шагов. Передвижение точки определяется специальными формулами и зависит от её текущего положения (i,j), динамического параметра l0 в нулевой момент времени (k=0) подаются со стандартного потока ввода. Результатом работы программы является информация о попадании точки в заданную геометрическую область. Вычисления останавливаются при попадании точки в заданную область или при превышении допустимого количества шагов.

Геометрическая область: кольцо, ограниченное двумя окружностями с центром в точке (10,10), радиус внутренней окружности равен 5, а радиус внешней равен 10.

Формулы для вычисления координат и параметра:

$i0 = -1, j0 = -1, l0 = -9$

$ik+1 = \max(jk-k, lk-k) \bmod 30 + \max(ik+lk, jk+k) \bmod 20$

$jk+1 = (|ik-lk| * \text{sign}(jk+k) + |ik-k| * (jk+k)) \bmod 20$

$lk+1 = (ik+k) * (jk-k) * (lk+k) \bmod 25$

4. Оборудование (лабораторное):

ЭВМ _____, процессор _____, имя узла сети _____ с ОП _____ Мб,
НМД _____ Мб. Терминал _____ адрес _____. Принтер _____
Другие устройства _____

Оборудование ПЭВМ студента, если использовалось:

Процессор 2,9 GHz 2-ядерный процессор Intel Core i5 с ОП 8 Гб, НМД 500 Гб. Монитор 13,3-дюймовый (2560 x 1600).

Другие устройства _____

5. Программное обеспечение (лабораторное):

Операционная система семейства _____, наименование _____ версия _____
интерпретатор команд _____ версия _____

Система программирования _____ версия _____

Редактор текстов _____ версия _____

Утилиты операционной системы _____

Местонахождение и имена файлов программ и данных _____

Программное обеспечение ЭВМ студента, если использовалось:

Операционная система семейства UNIX, наименование Terminal версия 2.10

интерпретатор команд bash версия 3.2.

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

Лабораторная работа №8:

Изучим работу компилятора GNU Compiler Collection 12.0.0, отладив простейшую программу – подсчет количества чётных цифр в числе. Посмотрим на указания компилятора к допущенным ошибкам – отсутствие закрывающей скобки в конце программы, отсутствие точки с запятой в конце строки, не взятое в скобки условие после if, отсутствие объявления переменной, отсутствие подключения основной библиотеки.

Лабораторная работа №9:

$$i0 = -1, j0 = -1, l0 = -9$$

$$i1 = i0 + 1 = \max(j0 - 0, l0 - 0) \bmod 30 + \max(i0 + l0, j0 + 0) \bmod 20 = \max(-1 - 0, -9 - 0) \bmod 30 + \max(-1 - 9, -1 - 0) \bmod 20 = \max(-1, -9) \bmod 30 + \max(-10, -1) \bmod 20 = (-1) \bmod 30 + (-1) \bmod 20 = -1 - 1 = -2$$

$$j1 = j0 + 1 = (|i0 - l0| * \text{sign}(j0 + 0) + |i0 - 0| * (j0 + 0)) \bmod 20 = (|-1 + 9| * \text{sign}(-1 + 0) + |-1 - 0| * (-1 + 0)) \bmod 20 = (8 * \text{sign}(-1) + |-1| * (-1)) \bmod 20 = (8 * (-1) + 1 * (-1)) \bmod 20 = (-8 - 1) \bmod 20 = (-9) \bmod 20 = -9$$

$$l1 = l0 + 1 = (i0 + 0) * (j0 - 0) * (l0 + 0) \bmod 25 = (-1 + 0) * (-1 - 0) * (-9 + 0) \bmod 25 = (-1) * (-1) * (-9) \bmod 25 = (-9) \bmod 25 = -9$$

$$i2 = i1 + 1 = \max(j1 - 1, l1 - 1) \bmod 30 + \max(i1 + l1, j1 + 1) \bmod 20 = \max(-9 - 1, -9 - 1) \bmod 30 + \max(-2 + (-9), -9 + 1) \bmod 20 = \max(-10, -10) \bmod 30 + \max(-11, -8) \bmod 20 = (-10) \bmod 30 + (-8) \bmod 20 = -10 + (-8) = -18$$

$$j2 = j1 + 1 = (|i1 - l1| * \text{sign}(j1 + 1) + |i1 - 1| * (j1 + 1)) \bmod 20 = (|-2 - (-9)| * \text{sign}(-9 + 1) + |-2 - 1| * (-9 + 1)) \bmod 20 = (7 * \text{sign}(-8) + |-3| * (-8)) \bmod 20 = (7 * (-1) + 3 * (-8)) \bmod 20 = (-7 + (-24)) \bmod 20 = (-31) \bmod 20 = -11$$

$$l2 = l1 + 1 = (i1 + 1) * (j1 - 1) * (l1 + 1) \bmod 25 = (-2 + 1) * (-9 - 1) * (-9 + 1) \bmod 25 = ((-1) * (-10) * (-8)) \bmod 25 = (-80) \bmod 25 = -5$$

$$i3 = i2 + 1 = \max(j2 - 2, l2 - 2) \bmod 30 + \max(i2 + l2, j2 + 2) \bmod 20 = \max(-11 - 2, -5 - 2) \bmod 30 + \max(-18 + (-5), -11 + 2) \bmod 20 = \max(-13, -7) \bmod 30 + \max(-23, -9) \bmod 20 = (-7) \bmod 30 + (-9) \bmod 20 = -7 + (-9) = -16$$

$$j3 = j2 + 1 = (|i2 - l2| * \text{sign}(j2 + 2) + |i2 - 2| * (j2 + 2)) \bmod 20 = (|-18 - (-5)| * \text{sign}(-11 + 2) + |-18 - 2| * (-11 + 2)) \bmod 20 = (13 * \text{sign}(-9) + |-20| * (-9)) \bmod 20 = (13 * (-1) + 20 * (-9)) \bmod 20 = (-13 + (-180)) \bmod 20 = (-193) \bmod 20 = -13$$

$$l3 = l2 + 1 = (i2 + 2) * (j2 - 2) * (l2 + 2) \bmod 25 = (-18 + 2) * (-11 - 2) * (-5 + 2) \bmod 25 = ((-16) * (-13) * (-3)) \bmod 25 = (-624) \bmod 25 = -24$$

$$i4 = i3 + 1 = \max(j3 - 3, l3 - 3) \bmod 30 + \max(i3 + l3, j3 + 3) \bmod 20 = \max(-13 - 3, -24 - 3) \bmod 30 + \max(-16 + (-24), -13 + 3) \bmod 20 = \max(-16, -27) \bmod 30 + \max(-40, -10) \bmod 20 = (-16) \bmod 30 + (-10) \bmod 20 = -16 + (-10) = -26$$

$$j4 = j3 + 1 = (|i3 - l3| * \text{sign}(j3 + 3) + |i3 - 3| * (j3 + 3)) \bmod 20 = (|-16 - (-24)| * \text{sign}(-13 + 3) + |-16 - 3| * (-13 + 3)) \bmod 20 = (8 * \text{sign}(-10) + |-19| * (-10)) \bmod 20 = (8 * (-1) + 19 * (-10)) \bmod 20 = (-8 + (-190)) \bmod 20 = (-198) \bmod 20 = -18$$

$$l4 = l3 + 1 = (i3 + 3) * (j3 - 3) * (l3 + 3) \bmod 25 = (-16 + 3) * (-13 - 3) * (-24 + 3) \bmod 25 = ((-13) * (-16) * (-21)) \bmod 25 = (-4368) \bmod 25 = -18$$

$$i5 = i4 + 1 = \max(j4 - 4, l4 - 4) \bmod 30 + \max(i4 + l4, j4 + 4) \bmod 20 = \max(-18 - 4, -18 - 4) \bmod 30 + \max(-26 + (-18), -18 + 4) \bmod 20 = \max(-22, -22) \bmod 30 + \max(-44, -14) \bmod 20 = (-22) \bmod 30 + (-14) \bmod 20 = -22 + (-14) = -36$$

$$j5 = j4 + 1 = (|i4 - l4| * \text{sign}(j4 + 4) + |i4 - 4| * (j4 + 4)) \bmod 20 = (|-26 - (-18)| * \text{sign}(-18 + 4) + |-26 - 4| * (-18 + 4)) \bmod 20 = (8 * \text{sign}(-14) + |-30| * (-14)) \bmod 20 = (8 * (-1) + 30 * (-14)) \bmod 20 = (-8 + (-420)) \bmod 20 = (-428) \bmod 20 = -8$$

$$l5 = l4 + 1 = (i4 + 4) * (j4 - 4) * (l4 + 4) \bmod 25 = (-26 + 4) * (-18 - 4) * (-18 + 4) \bmod 25 = ((-22) * (-22) * (-14)) \bmod 25 = (-6776) \bmod 25 = -1$$

$$i6 = i5 + 1 = \max(j5 - 5, l5 - 5) \bmod 30 + \max(i5 + l5, j5 + 5) \bmod 20 = \max(-8 - 5, -1 - 5) \bmod 30 + \max(-36 + (-1), -8 + 5) \bmod 20 = \max(-13, -6) \bmod 30 + \max(-37, -3) \bmod 20 = (-6) \bmod 30 + (-3) \bmod 20 = -6 + (-3) = -9$$

$$j6 = j5 + 1 = (|i5 - l5| * \text{sign}(j5 + 5) + |i5 - 5| * (j5 + 5)) \bmod 20 = (|-36 - (-1)| * \text{sign}(-8 + 5) + |-36 - 5| * (-8 + 5)) \bmod 20 = (35 * \text{sign}(-3) + |-41| * (-3)) \bmod 20 = (35 * (-1) + 41 * (-3)) \bmod 20 = (-35 + (-123)) \bmod 20 = (-158) \bmod 20 = -18$$

$$l6 = l5 + 1 = (i5 + 5) * (j5 - 5) * (l5 + 5) \bmod 25 = (-36 + 5) * (-8 - 5) * (-1 + 5) \bmod 25 = ((-31) * (-13) * 4) \bmod 25 = 1612 \bmod 25 = 12$$

k	i	j	l
0	-1	-1	-9
1	-2	-9	-9
2	-18	-11	-5

3	-16	-13	-24
4	-26	-18	-18
5	-36	-8	-1
6	-9	-18	12

Привожу вывод программы как доказательство того, что значения, посчитанные вручную и программой совпадают.

MacBook-Pro-MacBook:~ macbookpro\$ gcc lab9p.c -o fp.out

MacBook-Pro-MacBook:~ macbookpro\$./fp.out

-1 -1 -9

0 -1 -1 -9

1 -2 -9 -9

2 -18 -11 -5

3 -16 -13 -24

4 -26 -18 -18

5 -36 -8 -1

6 -9 -18 12

7 9 -1 21

8 24 4 -9

Попадание в заданную область не более чем за 50 шагов

Время попадания = 9

Конечные координаты точки = 11 5

Значение динамического параметра движения = 3

MacBook-Pro-MacBook:~ macbookpro\$

Алгоритм решения задачи:

- 1) Подключаем основную библиотеку stdio.h;
- 2) Прописываем функцию max (максимальное число из пары чисел): если x больше y, то возвращаем значение x, иначе – значение y;
- 3) Прописываем функцию sign (сигнум числа): если x больше нуля, то возвращаем единицу, иначе, если x меньше нуля, то возвращаем минус один, иначе – ноль;
- 4) Прописываем abs (модуль числа): если x больше или равен нулю, то возвращаем значение x, иначе – значение x, умноженное на минус единицу;
- 5) Прописываем функцию check (проверка координаты на принадлежность кольцу): если $(x-10) * (x-10) + (y-10) * (y-10)$ больше или равно 25 и меньше или равно 100, то возвращаем единицу, иначе – ноль. Примечание: мы отнимаем от x и y десять, так как центр окружностей, задающих кольцо, имеет координаты (10, 10). Число 25 появляется как квадрат радиуса меньшей из окружностей, а 100 – большей из окружностей;
- 6) Прописываем необходимые переменные: i, j, l, il, jl, ll, k=0, n=50;
- 7) На ввод подаем три переменные: i, j, l;
- 8) Исполняем цикл, пока k меньше n и функция check (i, j) возвращает ноль, прибавляя каждый раз к k единицу:
 1. Вычисляем il, jl, ll по заданным в варианте формулам;
 2. Переменным i, j, l присваиваются значения il, jl, ll соответственно;
- 9) После выхода из цикла проверяем функцию check (i, j). Если возвращена единица, то выводим на экран Yes и на следующей строке значения i, j, l, k. Если возвращён ноль, то выводим на экран No и на следующей строке значения i, j, l, n.

Будет использоваться компилятор GNU Compiler Collection 12.0.0.

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

Лабораторная работа №8:

1) Ошибка в отсутствии закрывающей скобки (}) в конце программы:

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

```
int main() {
int x,k=0;
scanf("%d",&x);
while(x!=0){
if ((x%10)%2==0){
k+=1;}
x/=10;}
printf("%d\n",k);
```

2) Ошибка в отсутствии точки с запятой (;) после k+=1:

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

```
int main() {
int x,k=0;
scanf("%d",&x);
```

```
while(x!=0){
if ((x%10)%2==0){
k+=1;
x/=10;}
printf("%d\n",k);}
```

3) Ошибка – условие после if не взято в скобки:

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

```
int main() {
int x,k=0;
scanf("%d",&x);
while(x!=0){
if (x%10)%2==0 {
k+=1;}
x/=10;}
printf("%d\n",k);}
```

4) Ошибка – не объявлена переменная x:

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

```
int main() {
int k=0;
scanf("%d",&x);
while(x!=0){
if ((x%10)%2==0){
k+=1;}
x/=10;}
printf("%d\n",k);}
```

5) Ошибка – не подключена библиотека:

```
int main() {
int k=0;
scanf("%d",&x);
while(x!=0){
if ((x%10)%2==0){
k+=1;}
x/=10;}
printf("%d\n",k);}
```

6) Без ошибок:

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

```
int main() {
int x,k=0;
scanf("%d",&x);
while(x!=0){
if ((x%10)%2==0){
k+=1;}
x/=10;}
printf("%d\n",k);}
```

Тесты:

- 1) 1234
Вывод: 2
- 2) 12
Вывод: 1
- 3) 1197
Вывод: 0

Лабораторная работа №9:

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

```
int max(int x, int y) { return (x > y) ? x : y;}
```

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

```
int abs(int x){
if (x>=0) return x;
else return x*(-1);}
```

```

int check(int x, int y) {
if (((x - 10) * (x - 10) + (y - 10) * (y - 10) >= 25) && ((x - 10) * (x - 10) + (y - 10) * (y - 10) <= 100)) return 1;
else return 0;}

int main() {
int i, j, l, i1, j1, l1, n=50, k=0;
scanf("%d%d%d", &i, &j, &l);

for (k = 0; k < n && check(i,j)==0; k++) {
printf("%d %d %d %d\n", k, i, j, l);
i1 = (max(j-k, l-k))%30 + (max(i+1, j+k))%20;
j1 = (abs(i-l) * sign(j+k) + abs(i-k) * (j+k))%20;
l1 = ((i+k) * (j-k) * (l+k))%25;
i = i1; j = j1; l = l1;}

if (check(i, j)) { printf("Попадание в заданную область не более чем за 50 шагов\n");
printf("Время попадания = %d\n", k);
printf("Конечные координаты точки = %d %d\n", i, j);
printf("Значение динамического параметра движения = %d\n", l);}
else { printf("Промаш\n");
printf("Время окончания движения = %d\n", n);
printf("Конечные координаты точки = %d %d\n", i, j);
printf("Значение динамического параметра движения = %d\n", l);} }

```

Тест:

1) -1 -1 -9

Вывод: Попадание в заданную область не более чем за 50 шагов

Время попадания = 9

Конечные координаты точки = 11 5

Значение динамического параметра движения = 3

Пункты 1-7 отчета составляются строго до начала лабораторной работы.

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

Лабораторная работа №8:

Last login: Wed Nov 3 08:29:08 on ttys000

The default interactive shell is now zsh.

To update your account to use zsh, please run `chsh -s /bin/zsh`.

For more details, please visit <https://support.apple.com/kb/HT208050>.

MacBook-Pro-MacBook:~ macbookpro\$ cat zag.txt

```
*****
* Минеева Светлана Алексеевна *
*      М8О-105Б-21      *
*      Лабораторная работа №8      *
*****
```

MacBook-Pro-MacBook:~ macbookpro\$ ls -l |tail -10

```
-rwxr-xr-x  1 macbookpro  staff  49640  5 ноя 13:27 fl.out
-rw-r--r--  1 macbookpro  staff   122  5 ноя 15:22 lab81.c
-rw-r--r--  1 macbookpro  staff   122  5 ноя 15:23 lab82.c
-rw-r--r--  1 macbookpro  staff   122  5 ноя 15:23 lab83.c
-rw-r--r--  1 macbookpro  staff   121  5 ноя 15:24 lab84.c
-rw-r--r--  1 macbookpro  staff   101  5 ноя 15:24 lab85.c
-rw-r--r--  1 macbookpro  staff   123  5 ноя 15:24 lab86.c
-rw-r--r--  1 macbookpro  staff   912  3 ноя 21:34 lab9.c
-rw-r--r--  1 macbookpro  staff  2895  7 июл  2020 pslog_20200707_123036.log
-rw-r--r--  1 macbookpro  staff   208  8 окт 16:26 zag.txt
```

MacBook-Pro-MacBook:~ macbookpro\$ cat lab81.c

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

```
int main() {
int x,k=0;
scanf("%d",&x);
while(x!=0){
if ((x%10)%2==0){
k+=1;}
x/=10;}
printf("%d\n",k);
MacBook-Pro-MacBook:~ macbookpro$ gcc lab81.c -o fl.out
lab81.c:10:16: error: expected '}'
printf("%d\n",k);
^
```

lab81.c:3:12: note: to match this '{'

```
int main() {
^
```

1 error generated.

MacBook-Pro-MacBook:~ macbookpro\$ cat lab82.c

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

```
int main() {
int x,k=0;
scanf("%d",&x);
while(x!=0){
if ((x%10)%2==0){
k+=1}
x/=10;}
printf("%d\n",k);}
MacBook-Pro-MacBook:~ macbookpro$ gcc lab82.c -o fl.out
lab82.c:8:5: error: expected ';' after expression
k+=1}
^
;
```

1 error generated.

```
MacBook-Pro-MacBook:~ macbookpro$ cat lab83.c
#include <stdio.h>
```

```
int main() {
int x,k=0;
scanf("%d",&x);
while(x!=0){
if (x%10)%2==0 {
k+=1;}
x/=10;}
printf("%d\n",k);}
```

```
MacBook-Pro-MacBook:~ macbookpro$ gcc lab83.c -o fl.out
```

lab83.c:7:10: error: expected expression

```
if (x%10)%2==0 {
^
```

1 error generated.

```
MacBook-Pro-MacBook:~ macbookpro$ cat lab84.c
#include <stdio.h>
```

```
int main() {
int k=0;
scanf("%d",&x);
while(x!=0){
if ((x%10)%2==0){
k+=1;}
x/=10;}
printf("%d\n",k);}
```

```
MacBook-Pro-MacBook:~ macbookpro$ gcc lab84.c -o fl.out
```

lab84.c:5:13: error: use of undeclared identifier 'x'

```
scanf("%d",&x);
^
```

lab84.c:6:7: error: use of undeclared identifier 'x'

```
while(x!=0){
^
```

lab84.c:7:6: error: use of undeclared identifier 'x'

```
if ((x%10)%2==0){
^
```

lab84.c:9:1: error: use of undeclared identifier 'x'

```
x/=10;}
^
```

4 errors generated.

```
MacBook-Pro-MacBook:~ macbookpro$ cat lab85.c
```

```
int main() {
int k=0;
scanf("%d",&x);
while(x!=0){
if ((x%10)%2==0){
k+=1;}
x/=10;}
printf("%d\n",k);}
```

```
MacBook-Pro-MacBook:~ macbookpro$ gcc lab85.c -o fl.out
```

lab85.c:3:1: error: implicitly declaring library function 'scanf' with type 'int (const char *restrict, ...)' [-Werror,-Wimplicit-function-declaration]

```
scanf("%d",&x);
^
```

lab85.c:3:1: note: include the header <stdio.h> or explicitly provide a declaration for 'scanf'

lab85.c:3:13: error: use of undeclared identifier 'x'

```
scanf("%d",&x);
^
```

lab85.c:4:7: error: use of undeclared identifier 'x'

```

while(x!=0){
^
lab85.c:5:6: error: use of undeclared identifier 'x'
if ((x%10)%2==0){
^
lab85.c:7:1: error: use of undeclared identifier 'x'
x/=10;}
^
lab85.c:8:1: error: implicitly declaring library function 'printf' with type
'int (const char *, ...)' [-Werror,-Wimplicit-function-declaration]
printf("%d\n",k);}
^
lab85.c:8:1: note: include the header <stdio.h> or explicitly provide a
declaration for 'printf'
6 errors generated.
MacBook-Pro-MacBook:~ macbookpro$ cat lab86.c
#include <stdio.h>

```

```

int main() {
int x,k=0;
scanf("%d",&x);
while(x!=0){
if ((x%10)%2==0){
k+=1;}
x/=10;}
printf("%d\n",k);}
MacBook-Pro-MacBook:~ macbookpro$ gcc lab86.c -o f1.out
MacBook-Pro-MacBook:~ macbookpro$ ./f1.out
1234
2
MacBook-Pro-MacBook:~ macbookpro$ ./f1.out
12
1
MacBook-Pro-MacBook:~ macbookpro$ ./f1.out
1197
0
MacBook-Pro-MacBook:~ macbookpro$

```

Лабораторная работа №9:

Last login: Wed Nov 3 11:37:45 on ttys000

The default interactive shell is now zsh.

To update your account to use zsh, please run `chsh -s /bin/zsh`.

For more details, please visit <https://support.apple.com/kb/HT208050>.

MacBook-Pro-MacBook:~ macbookpro\$ cat zag.txt

```

*****
* Минеева Светлана Алексеевна *
*      М8О-105Б-21             *
*  Лабораторная работа №9     *
*****

```

MacBook-Pro-MacBook:~ macbookpro\$ cat >f1.out

^C

MacBook-Pro-MacBook:~ macbookpro\$ cat >lab9.c

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

```
int max(int x, int y) { return (x > y) ? x : y;}
```

```

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

```



```

int abs(int x){
if (x>=0) return x;
else return x*(-1);}

int check(int x, int y) {
if (((x - 10) * (x - 10) + (y - 10) * (y - 10) >= 25) && ((x - 10) * (x - 10) + (y - 10) * (y - 10) <= 100)) return 1;
else return 0;}

int main() {
int i, j, l, il, j1, l1, n=50, k=0;
scanf("%d%d%d", &i, &j, &l);

for (k = 0; k < n && check(i,j)==0; k++) {
il = (max(j-k, l-k))%30 + (max(i+l, j+k))%20;
j1 = (abs(i-l) * sign(j+k) + abs(i-k) * (j+k))%20;
l1 = ((i+k) * (j-k) * (l+k))%25;
i = il; j = j1; l = l1;}

if (check(i, j)) { printf("Попадание в заданную область не более чем за 50 шагов\n");
printf("Время попадания = %d\n", k);
printf("Конечные координаты точки = %d %d\n", i, j);
printf("Значение динамического параметра движения = %d\n", l);}
else { printf("Промех\n");
printf("Время окончания движения = %d\n", n);
printf("Конечные координаты точки = %d %d\n", i, j);
printf("Значение динамического параметра движения = %d\n", l);}}
^C

```

```

MacBook-Pro-MacBook:~ macbookpro$ ls -l |tail -4
-rw-r--r--  1 macbookpro  staff   0 3 ноя 17:38 fl.out
-rw-r--r--  1 macbookpro  staff  866 3 ноя 17:40 lab9.c
-rw-r--r--  1 macbookpro  staff 2895 7 июл 2020 pslog_20200707_123036.log
-rw-r--r--  1 macbookpro  staff  208 8 окт 16:26 zag.txt
MacBook-Pro-MacBook:~ macbookpro$ cat lab9.c
#include <stdio.h>

```

```

int max(int x, int y) { return (x > y) ? x : y;}

```

```

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

```

```

int abs(int x){
if (x>=0) return x;
else return x*(-1);}

```

```

int check(int x, int y) {
if (((x - 10) * (x - 10) + (y - 10) * (y - 10) >= 25) && ((x - 10) * (x - 10) + (y - 10) * (y - 10) <= 100)) return 1;
else return 0;}

```

```

int main() {
int i, j, l, il, j1, l1, n=50, k=0;
scanf("%d%d%d", &i, &j, &l);

```

```

for (k = 0; k < n && check(i,j)==0; k++) {
il = (max(j-k, l-k))%30 + (max(i+l, j+k))%20;
j1 = (abs(i-l) * sign(j+k) + abs(i-k) * (j+k))%20;
l1 = ((i+k) * (j-k) * (l+k))%25;
i = il; j = j1; l = l1;}

```

```

if (check(i, j)) { printf("Попадание в заданную область не более чем за 50 шагов\n");
printf("Время попадания = %d\n", k);
printf("Конечные координаты точки = %d %d\n", i, j);
printf("Значение динамического параметра движения = %d\n", l);}
else { printf("Промех\n");
printf("Время окончания движения = %d\n", n);
printf("Конечные координаты точки = %d %d\n", i, j);
printf("Значение динамического параметра движения = %d\n", l);}}

```

```

MacBook-Pro-MacBook:~ macbookpro$ gcc lab9.c -o fl.out

```

```

MacBook-Pro-MacBook:~ macbookpro$ ./fl.out

```

```

-1 -1 -9

```

Попадание в заданную область не более чем за 50 шагов

Время попадания = 9

Конечные координаты точки = 11 5

Значение динамического параметра движения = 3

MacBook-Pro-MacBook:~ macbookpro\$

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
1	Дом.	03.11.21	11:00	Вывод неверных чисел	Ошибка исправлена, значения сначала присваиваются переменным i1, j1, l1, а после передаются переменным i, j, l	Нельзя сразу изменять значения переменных i, j, l, так как в нескольких следующих строках требуются изначальные значения

10. Замечания автора по существу работы

Данные работы очень полезны, они отлично развивают мышление.

11. Выводы

Лабораторная работа №8: Я изучила конкретную систему программирования на Си и получила навыки подготовки текстов и отладки программ. Лабораторная работа №9: Я составила и отладила простейшую программу на языке Си 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 .

Недочёты при выполнении задания могут быть устранены следующим образом: больше практиковаться в написании программ на языке Си.

Подпись студента Минеева С.А