

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

Студент группы М80-109Б-22 Степанов Алексей Николаевич, № по списку 18

Контакты aleksey.stepanov2004@mail.ru, telegram @Alex1stepa

Работа выполнена: «13» ноября 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

Отчет сдан « » _____ 20__ г., итоговая оценка _____

Подпись преподавателя _____

1. Тема: Программирование на языке С.

2. **Цель работы:** Составление и отладка простейшей программы на языке Си итеративного характера с целочисленными рекуррентными соотношениями, задающими регулярное некоторое движение точки в целочисленной системе координат (i, j) с дискретным временем k и динамическим параметром движения l.

Задание (вариант №14): Лунка, являющаяся пересечением двух кругов радиуса 10, центр первого круга – в точке (-10, -10), центр второго – в точке (-20, -20), $i_0=6$, $j_0=27$, $i_0=-15$...

3. Оборудование (студента):

Процессор *Intel Core i5-8265U @ 8x 3.9GHz* с ОП 7851 Мб, НМД 1024 Гб. Монитор 1920x1080

4. Программное обеспечение (студента):

Операционная система семейства: *linux*, наименование: *ubuntu*, версия *18.10 cosmic*

интерпретатор команд: *bash* версия *4.4.19*.

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

Утилиты операционной системы --

Прикладные системы и программы: VTM(QT)

Местонахождение и имена файлов программ и данных на домашнем компьютере --

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

1) К сожалению, функции для вычисления $k+1$ параметров используют k значения => мы будем использовать промежуточные переменные для хранения результатов вычисления первых двух (а затем и 3) значения $k+1$ параметра, чтобы не испортить вычисления;

2) Нам понадобятся функция modulo (в си у нас под % реализован remainder), а также функции минимума и максимума из 2 чисел, алгоритм описания которых интуитивен или (как с modulo) общеизвестен (например, его дают на онлайн курсах Сириуса) – не вижу смысла приводить, все равно в коде будет.

3) Лунка образована пересечением 2 окружностей >= точка должна быть в пересечении 2ух окружностей => по условию точка должна удовлетворять уравнениям $(i-20)^2 + (j-20)^2 \leq 100$, $(i-20)^2 + (j-20)^2 \leq 100$ (то есть меньше(внутри) или равна(на границе) окружностей).

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

Тесты (сделал в понедельник(7.11)):

k	i	j	l
0	6	27	-15
1	18	0	15
2	6	29	29
3	14	22	27
4	16	9	15
5	18	2	26
6	15	7	19
7	5	9	9
8	18	8	2
9	0	24	4
10	11	21	15
11	15	5	11
12	10	4	4
13	8	22	28

14	3	9	15
15	19	1	1
16	4	16	16
17	8	0	0
18	15	13	13
19	17	19	27
20	18	28	2
21	8	2	22
22	11	11	7
23	1	7	19
24	14	26	14
25	8	20	20
26	7	25	25
27	17	29	29
28	6	14	2
29	12	28	8
30	19	13	19
31	11	1	19
32	18	10	18
33	12	28	10
34	3	27	15
35	5	11	11
36	10	0	6
37	0	24	24
38	3	23	17
39	15	25	9
40	0	6	6
41	0	20	26
42	15	19	9
43	3	27	3
44	8	14	14
45	8	12	0
46	19	15	27
47	1	29	29
48	14	12	12
49	16	0	24
50	11	11	5

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

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

```
const int i0=6,j0=27,l0=-15;
```

```
int mod(int y,int md){
```

```
    return ( y % md + md) % md;
```

```
}
```

```
// функция mod находит modulo для целого числа y по целому числу md
```

```
int min_custom(int first_num,int second_num){
```

```
    if(first_num<second_num)
```

```
        return first_num;
```

```

else
    return second_num;

}
//функция min_custom находит минимальное из 2 целых чисел
int max_custom(int first_num,int second_num){
    if(first_num<second_num)
        return second_num;
    else
        return first_num;

}
//функция max_custom находит максимальное из 2 целых чисел
int main(){

int i=i0,j=j0,l=l0,ik1=0,jk1=0,lk1=0;
setlocale(LC_ALL, "Rus");//включим русский язык
for( int k=0; k<50; k++ ){

    ik1=mod( ( (i*i*i) - (j*j*j) + (l*l*l) - k) , 20);
    jk1=mod( min_custom( (i*j*j) - k, min_custom( (i*i*l) -k
        ,(l*l*j) -k)) , 30 );
    lk1=mod( max_custom( (i*j*j) - k, max_custom( (i*i*l) -k
        ,(l*l*j) -k)) , 30 );

    i=ik1;
    j=jk1;
    l=lk1;
    if ( ( (i+10) * (i+10) +(j+10)*(j+10) <=100 )&&
        ( (i+20) * (i+20) +(j+20)*(j+20) <=100 )){

        printf("точка достигнет лунку на шаге под номером %d, переменные при этом приняли значения : i=%d,
j=%d, l=%d \n",k,i,j,l);
        return 0; //зачем выходить из цикла, если можно вернуть значение 0 и убить процесс?

    }
}
printf("точка не достигается,к сожалению,переменные при этом приняли значения : i=%d, j=%d, l=%d \n",i,j,l);
return 0;
}
alex1st@alex1st-VirtualBox:~$ gcc task9.c
alex1st@alex1st-VirtualBox:~$ ./a.out
точка не достигается,к сожалению,переменные при этом приняли значения : i=11, j=11, l=5

```

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	Вре мя	Событие	Действие по исправлению	Примечание

10. Замечания автора по существу работы
Работа конструктивная, негативный замечаний нет.

11. Выводы

От лабораторной работы получил исключительно положительные эмоции и впечатления. По моему мнению, знания, приобретенные мною на данной лабораторной работе, помогли мне улучшить мой стиль написания кода на C++, укрепить мои знания и улучшить умения написания функции в Си, а также именования функций и переменных, подобно тому, как это делают в уважаемых и серьезных конторах.

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

Подпись студента _____