Образец оформления отчёта по Лабораторной работе.

|  |  |
| --- | --- |
| **К Г Э У** | МИНИСТЕРСТВО ВЫСШЕГО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ ЭНЕРГЕТИЧЕСКИЙ УНИВЕРСИТЕТ»**  (ФГБОУ ВО «КГЭУ») |

**Кафедра Информатики и информационных управляющих систем**

**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 8**

**РЕШЕНИЕ ЗАДАЧ С ОДНОМЕРНЫМИ СТАТИЧЕСКИМИ МАССИВАМИ МЕТОДАМИ СТРУКТУРНОГО ПРОГРАММИРОВАНИЯ. АНАЛИЗ КОДОВ ПРОГРАММ С ПЕРЕДАЧЕЙ ПАРАМЕТРОВ.**

|  |  |
| --- | --- |
| **Исполнитель:** | Банницин Дмитрий |
| **Группа:** | ТРП-3-22 |
| **Вариант:** | 35 |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Казань -2022

**Задания для самостоятельной работы**

**Задача №1**

Дан целый массив a[19], заполнить его по формуле a[i] = 7\*i+11, напечатать его в порядке возрастания количества бит установленных для a[i] в 0.

**Код**

#include <iostream>

using namespace std;

void fillMass(int mass[19]) {

for (size\_t i = 0; i < 19; i++)

{

mass[i] = 7 \* i + 11;

}

}

void getNumbers(int mass[19], int numbers[19]) {

int kbit; // описание типов переменных

for (size\_t i = 0; i < 19; i++)

{

kbit = 0;

for (int j = 0; j < 8 \* sizeof(int); j++) // цикл по битам

if ((mass[i] & (1L << j)) != 0)

kbit++;

numbers[i] = 32 - kbit;

}

}

void sortNumbers(int numbers[19], int mass[19]) {

for (size\_t i = 0; i < 18; i++)

{

for (size\_t j = 0; j < 18-i; j++)

{

if (numbers[j] > numbers[j + 1])

{

swap(numbers[j], numbers[j+1]);

swap(mass[j], mass[j + 1]);

}

}

}

}

int main()

{

int mass[19];

int numbers[19];

fillMass(mass);

getNumbers(mass, numbers);

cout << "mass: ";

for (size\_t i = 0; i < 19; i++)

{

cout << mass[i] << " ";

}

cout << endl;

cout << "null bit count in mass: ";

for (size\_t i = 0; i < 19; i++)

{

cout << numbers[i] << " ";

}

cout << "\n-----------------------------------------------------";

sortNumbers(numbers, mass);

cout << endl;

cout << "sorted null bit count in mass: ";

for (size\_t i = 0; i < 19; i++)

{

cout << numbers[i] << " ";

}

cout << endl;

cout << "sorted mass: ";

for (size\_t i = 0; i < 19; i++)

{

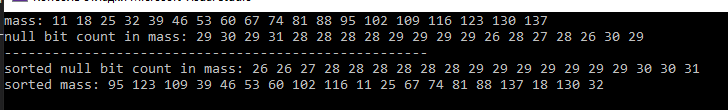
cout << mass[i] << " ";

}

cout << endl;

}

**Решение**

****

**Задача №2**

Изучить текст программы и показать, что будет выведено на консоль, если на запрос с консоли ввести x=0.1 и eps=0.05. Показать и объяснить промежуточные результаты.

**Код**

#include<iostream>

#include<cmath>

using namespace std;

float d(float x, float e);

int main()

{

float x, eps, z;

cout << "\nInput argument << 1:"; cin >> x;

cout << "\nInput accuracy:"; cin >> eps;

z = d(x, eps);

cout << "\nFor x = " << x << " z(x) = " << z << endl;

system("pause");

return 0;

}

float d(float x, float e)

{

float sum = 1., u = 2., cf = -1;

for (int i = 1; abs(u) > e; i++)

{

u \*= x \* x;

sum += cf \* u;

cf = -cf;

cout << "\n i = " << i << endl;

}

return sum;

}

**Решение**

Основыной частью работы программы является работа функции «d», принимающей на вход число, которое будет преобразовано по формуле

И точность результата.

Преобразование производится с помощью цикла в котором каждую итерацию исходное число возводится в квадрат, домножается на 2 и прибавляется к сумме ряда с очередным коэффициентом.

Так же каждую итерацию выводится ее номер, а в конце главной функции результат вычислений.

Если на запрос с консоли ввести x=0.1 и eps=0.05, будет выведено следущее:

i = 1

For x = 0.1 z(x) = 0.98

Тоесть выполнится 1 итерация, при которой переменные примут значения:

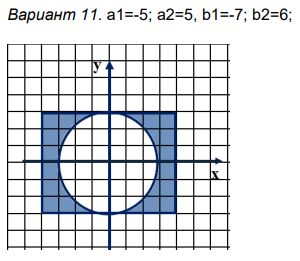
u = 2 \* 0.1 \* 0.1 = 0.02

sum = 1 - 0.02 = 0.98

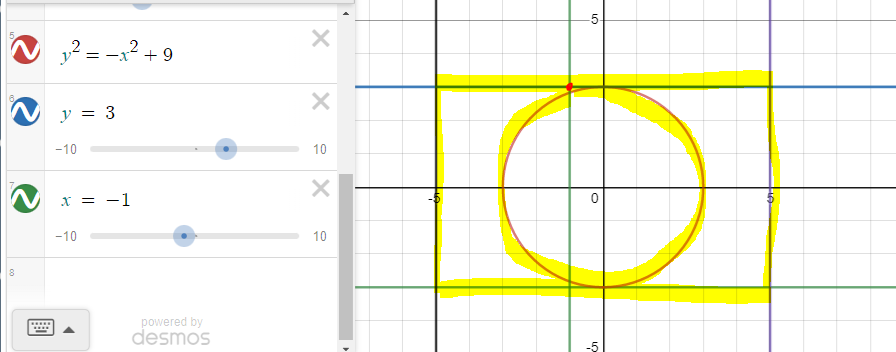
cf = -(-1) = 1

**Задача №3**

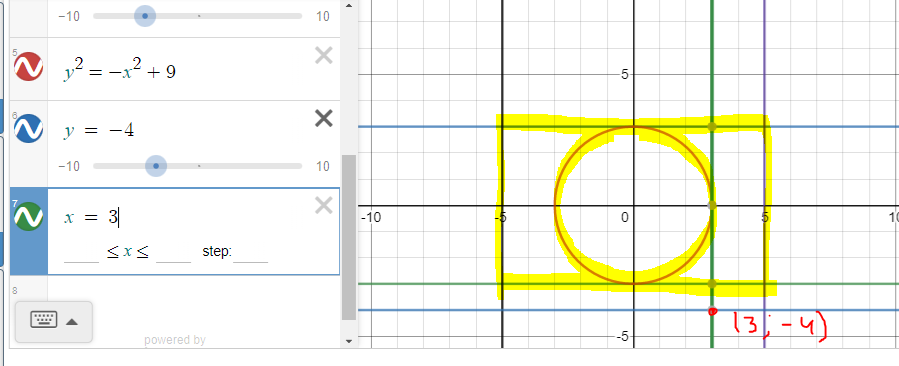
По «мишени», изображенной на рисунке (закрашенная фигура), происходит обстрел шариками, которые случайным образом попадают в поле по x∈[a1; a2] и по y∈[b1; b2]. Составить две функции, одна генерирует координаты шариков в заданном диапазоне, а вторая определяет, попал шарик в мишень или не попал.



**Контрольный пример**



(-1;3)



**Код**

#include <iostream>

using namespace std;

void getCoords(int& x, int& y);

bool isHit(int x, int y);

int main()

{

int a1 = -5;

int a2 = 5;

int b1 = -7;

int b2 = 7;

int x, y;

getCoords(x, y);

cout << "0 - Shot\n";

cout << "1 - Stop\n\n";

int action;

cin >> action;

while (action != 1)

{

getCoords(x, y);

if (isHit(x, y))

{

cout << "hit!\n";

cout << "x = " << x << " y = " << y << endl;

cout << "--------\n";

}

else

{

cout << "miss!\n";

cout << "x = " << x << " y = " << y << endl;

cout << "--------\n";

}

cout << "0 - Shot\n";

cout << "1 - Stop\n\n";

cin >> action;

}

}

void getCoords(int& x, int& y) {

x = rand() % 10 - 5;

y = rand() % 15 - 7;

}

bool isHit(int x, int y) {

if (sqrt(pow((float)x, 2) + pow(y, 2)) >= 3 && x <= 4 && y <= 3 && x >= -4 && y >= -3)

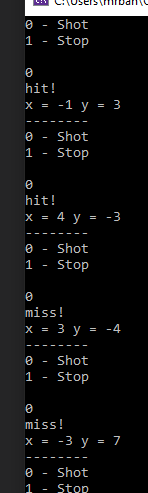
return true;

else

return false;

}

**Решение**

****

**Задача №4**

Заголовок функции имеет вид: void P(int& a, char\* b, float c); Переменные в вызывающей функции описаны так: int a; char\* b, \*c; float d,x; Выбрать правильные варианты и обосновать, почему отвергаются оставшиеся варианты.

Вариант 1 P(a,b,c);

Вариант 2 P(3,b,x);

Вариант 3 P(a,b+5,d-a);

Вариант 4 P(a,c,d-x/2);

**Решение**

Правильными вариантами являются 3 и 4.

Вариант 1 не подходит, поскольку 3-й аргумент является указателем на переменную символьного типа, в то время как 3-м параметром вызываемой функции является вещественное число.

Вариант 2 не подходит, поскольку 1-й аргумент является константным значением и не может быть передан в качестве адреса;

**Домашнее задание.**

**Задание №1.**

Изучить текст программы и показать, что будет выведено на консоль, показав и объяснив промежуточные результаты.

**Код**

#include <iostream>

using namespace std;

//Объявление (прототипы) функций:

void tab(int b, int e, int s);

float res1(int x, int st);

float res2(int x, int st);

int main()

{

int xn = -8, xk = 8, dx = 4;

cout << "\n Construction table of selected function\n";

tab(xn, xk, dx);

system("pause");

return 0;

}

void tab(int b, int e, int s)

{

for (int i = b; i <= e; i += s)

{

float result;

if (i < 0) result = res1(i, 2);

else if (i > 0) result = res2(i, 3);

else result = 0;

cout << "\t" << i << "\t" << result << endl;

}

return;

}

float res1(int x, int st)

{

return x / st;

}

float res2(int x, int st)

{

return (float)x / st;

}

**Решение**

* Из главной программы вызывается функция “Tab”, в которую передается 3 аргумента -8; 8 и 4.
* В вызванной функции работает цикл пока i <= e (при их начльных значениях i = -8 и e = 8), шаг цикла = 4. Таким образом будет выполнено 5 итераций, при которых i примет значения -8, -4, 0, 4 и 8 после чего цикл прекратится.
* Внутри цикла каждую итерацию происходит ветвление, в зависимости от значения переменной i, и выводятся значения i и result.

1. При i < 0, вызывается функция, возвращающая целочисленный результат деления значения переменной i на 2, который будет помещен в переменную result.
2. При i > 0, вызывается функция, возвращающая вещественный результат деления значения переменной i на 3, который будет помещен в переменную result..
3. При i = 0, result будет равен 0.

* Таким образом в результате работы будет выведено:

-8 -4

-4 -2

0 0

4 1.33333

8 2.66667