Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

Лабораторная работа №8

По дисциплине «Основы алгоритмизации и программирования»

На тему «Вычисление сумм, произведений, экстремумов»

Выполнил:

Студент 1 курса 7 группы

Ленкевич Павел Андреевич

Преподаватель: асс. Андронова М.В.

2023, Минск

**Вариант 9**

| **№** | **Формулы для вычислений** | **Исходные данные** |
| --- | --- | --- |
| 9 |  |  |
|  |  |  |

**Задание 1.**

#include <iostream> //Подключение библиотеки

using namespace std; // Позволяет не писать std перед потоковыми операторами Ввода-вывода

float main() // Тело программы

{

setlocale(LC\_CTYPE, "Russian"); // Подключает русский язык

float d, x, q=0, t = 0.45; //3 Задаём переменные и их значения

for (int i = 0; i < 6; i++) { // Цикл с шестью повторами

cout << "Введите x: "; cin >> x; // Ввод значения переменной x

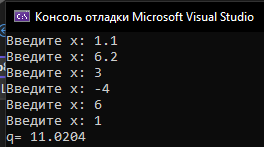
d = t + (x + 1) / x; // Расчёт дополнительной переменной d

q += d; // Расчёт значения q

}

cout << "q= " << q; // Вывод значения q

}

****

**Задание 2.**

#include <iostream> // Подключение библиотек

#include <algorithm>

using namespace std; // Позволяет не писать std перед каждым оператором

float main() // Тело программы

{

setlocale(LC\_CTYPE, "Russian"); // Подключает русский язык

float p = 9999, q = 1, y; // Задаём переменные и их значения

for (int i = 0; i < 5; i++) { // Цикл с количеством повторов

cout << "Введите y: "; cin >> y; // Вводим значение переменной

p = min(p, y); // Выбирает минимальное значение среди p и y и присваивает его переменной p

q = q \* (y - 5); // Расчёт q - произведений переменных

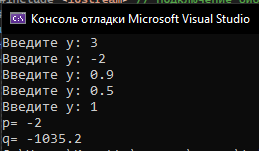
}

q += p; // Финальный расчёт q

cout << "p= " << p << endl; // Вывод значений переменных

cout << "q= " << q;

}



**Вариант 1.**

****

**Задание 1.**

#include <iomanip> //подключает библиотеки

#include <iostream>

using namespace std; // Позволяет не писать std перед каждым оператором

float main() // Тело программы

{

setlocale(LC\_CTYPE, "Russian"); // Подключает русский язык

int n = 5; // Задаёт переменные и их значения

float d = 12.5 \* pow(10, -4), a, h, sum=0;

for (int i = 0; i < n; i++) // Цикл на n повторов

{

cout << "a = "; cin >> a; // Ввод переменной а

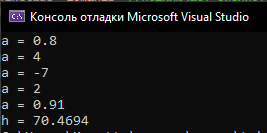
sum += a\*a; // Вычисление значения sum

}

h = d + sum; // Финальное значение sum

cout << "h = " << h; // Вывод значения переменной

}

****

**Задание 2.**

#include <iomanip> //подключает библиотеки

#include <iostream>

#include <cmath>

#include <algorithm>

using namespace std; // Позволяет не писать std перед каждым оператором

float main() // Тело программы

{

setlocale(LC\_CTYPE, "Russian"); // Подключает русский язык

float x, z, y=0; // Задаёт переменные и их значения

for (int i = 0; i < 5; i++) // Цикл на 5 повторов

{

cout << "Введите x: "; cin >> x; // Ввод значения переменной x

z = max(z, x); // Наъодим максимальный x и приравниваем его к z

y += x \* x; // Вычисление значения переменной y

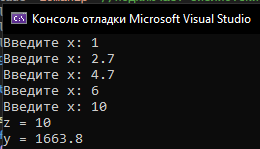
}

y = z \* y; // Финальное вычисление значения y

cout << "z = " << z << endl; // Вывод значений переменных

cout << "y = " << y;

}

****

**Дополнительные задания**

**Задание 1.**

В последовательности из **n** целых чисел найти и вывести значение суммы четных элементов.

#include <iostream> //подключает библиотеки

using namespace std; // Позволяет не писать std перед каждым оператором

int main() // Тело программы

{

setlocale(LC\_CTYPE, "Russian"); // Подключает русский язык

int sum = 0, n; // Задаём переменные и их значения

cout << "Введите количество элементов (n): "; cin >> n; //вводим количество элементов последовательности

for (int i = 0; i < n; i++) { // Цикл на n повторов

if (i % 2 == 0) { // Если при делении i на 2 в остатке будет 0, то изменится значение переменной sum

sum += i;

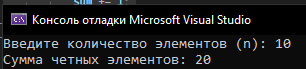
}

}

cout << "Сумма четных элементов: " << sum; // Вывод значения переменной

return 0; // Приравнивает переменные к 0, для корректного последующего запуска

}

****

**Задание 2.**

В последовательности из **n** целых чисел найти и вывести порядковый номер последнего отрицательного элемента.

#include <iostream> // Подключение библиотеки

using namespace std; // Позволяет не писать std перед каждым оператором

int main() // Тело программы

{

setlocale(LC\_CTYPE, "Russian"); // Подключает русский язык

int a, n, nomer; // Задаём переменные и их значения

cout << "Введите количество элементов (n): "; cin >> n; // Вводим количество элементов последовательности

for (int i = 1; i <= n; i++) { // Цикл на n повторов

cout << "Введите число номер "<<i<<": "; cin >> a; // Ввод переменной а

if (a < 0) { // Если введённое число < 0, то переменной nomer присвоится значение i

nomer = i;

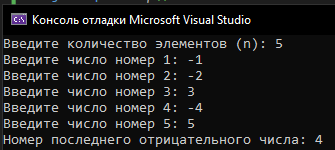
}

}

cout << "Номер последнего отрицательного числа: " << nomer; // Вывод значения перменной

return 0; // Приравнивает переменные к 0, для корректного последующего корректного запуска

}

****

**Задание 3.**

В последовательности из **n** вещественных чисел найти количество элементов, стоящих между минимальным и максимальным значениями.

#include <iostream> // Подключение библиотеки

#include <algorithm>

using namespace std; // Позволяет не писать std перед каждым оператором

int main() // Тело программы

{

setlocale(LC\_CTYPE, "Russian"); // Подключает русский язык

double minx, maxx, x, posminx = 0, posmaxx = 0; //Задаём переменные и их значения

int n;

cout << "Введите количество элементов: "; cin >> n; // Присваиваем значение переменной n

for (int i = 1; i <= n; i++) // Цикл с числом повторов n

{

cout << "Введите элемент номер "<<i<<": "; cin >> x; // Присваиваем значение переменной x

if (i == 1) { // Условие для первого круга цикла, если оно не выполнится, то это значит, что круг уже не первый

minx = x; // Приравнивание переменных

maxx = x;

posminx = i;

posmaxx = i;

}

else { // Для последующих кругов

if (x < minx) { // Если x<minx, то он и есть минимум и его позиция равна i

minx = x;

posminx = i;

}

else

if (x > maxx) { // Если x>maxx, то он и есть максимум и его позиция равна i

maxx = x;

posmaxx = i;

}

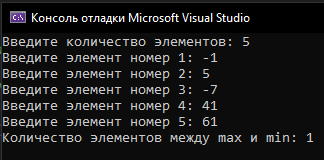
}

}

cout << "Количество элементов между max и min: " << abs(posmaxx - posminx) - 1 << endl; // Вывод модуля разности позиций -1, таким образом выводится количество элементов между max и min

return 0;

}

****