Санкт-Петербургский исследовательский университет

информационных технологий, механики и оптики

Кафедра программных систем

**Практическая работа №2.**

**Реализация управляющих операторов.**

Выполнила:

Анисимова Ксения Алексеевна

Группа: К3120

Проверила: Филимонова И. А.

Санкт-Петербург

2019

**Цель работы:** научиться реализовывать управляющие операторы.

**Задачи:**

1) Научиться реализовывать операторы выбора.

2) Научиться использовать циклы при реализации алгоритмов.

3) Научиться писать программу, производящую расчет суммы чисел на заданном интервале.

4) Научиться писать программу, имитирующую стрельбу по мишени.

**Ход работы:**

1.1 Создан алгоритм (Рисунок 1), проверяющий принадлежность точки заданной фигуре с помощью if-else-if.

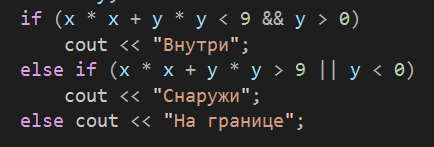


Рисунок 1 - Алгоритм

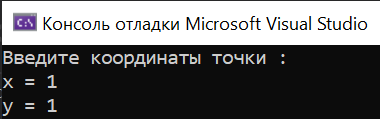


Рисунок 2 – Результат работы программы

1.2 Создан алгоритм (Рисунок 3), проверяющий, является ли данный год високосным с помощью конструкции if-else if.

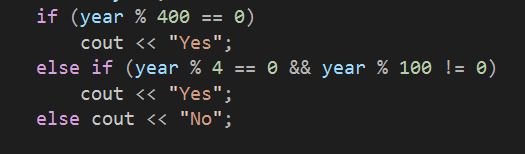


Рисунок 3 - Проверка високосный ли год

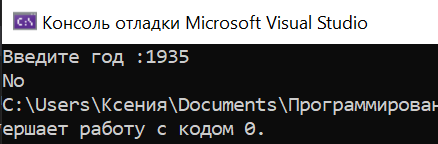


Рисунок 4 – Результат работы программы

1.3 Создана программа, в которой пользователь выбирает параметры автомобиля. Алгоритм был реализован с помощью оператора switch.

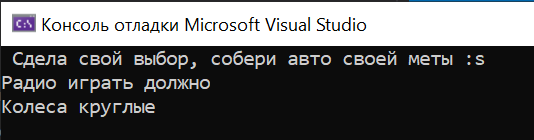


Рисунок 5 – Результат работы программы

2.1 С помощью цикла с постусловием организован алгоритм, который выводит на экран все значения синуса на указанном интервале с шагом 0,1.

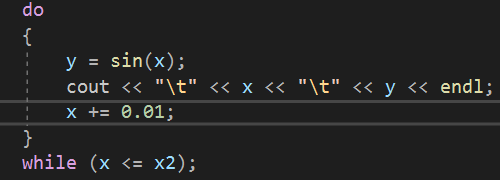


Рисунок 6 - Алгоритм вывода значений синуса на интервале

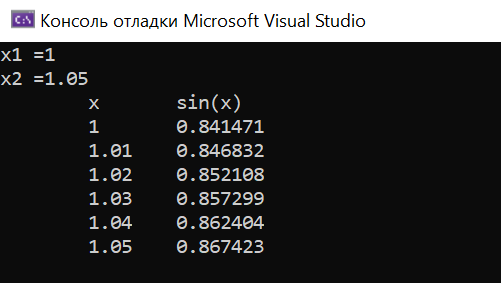


Рисунок 7 – Результат работы программы

2.2 Реализована программа, вычисляющая наибольший общий делитель двух чисел по алгоритму Евклида с использованием оператора с предусловием.

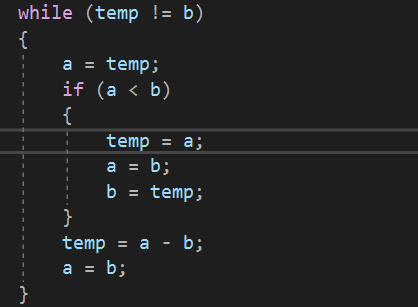


Рисунок 8 - Алгоритм Евклида

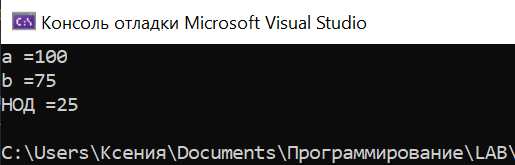


Рисунок 9 – Результат работы программы

2.3 Для программы, которая выводит на экран все значения синуса на указанном интервале с шагом 0,1 использован оператор с предусловием, а для программы, вычисляющей НОД по алгоритму Евклида, использован оператор с постусловием. В первом случае изменение оператора не повлияло на работу программы, а во втором случае для корректной работы программы потребовалось добавить дополнительное условие для выхода из цикла (Рисунок 10).

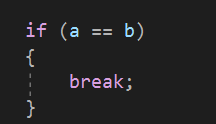


Рисунок 10 - Дополнительное условие

3.1 Создана программа для суммирования всех чисел в диапазоне от 1 до k и от m до 100. Для суммирования используется цикл for и оператор перехода continue (Рисунок 11).

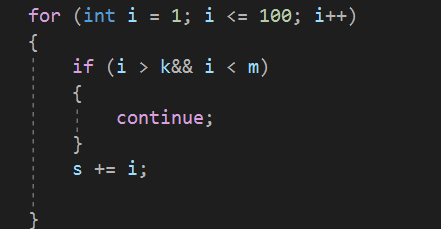


Рисунок 11 - Алгоритм, суммирующий числа в заданном диапазоне

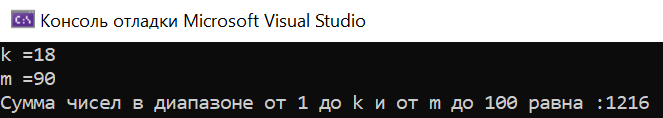


Рисунок 12 – Результат работы программы

4.1 Создана программа, которая определяет уровень мастерства стрелка. Пользователь вводит координаты точек (выстрелов), а программа подсчитывает баллы за попадание в определённую часть мишени (Рисунок 13).

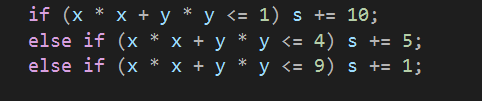


Рисунок 13 - Балльная оценка выстрелов

Далее программа в зависимости от процента попаданий выставляет свою оценку (Рисунок 14).

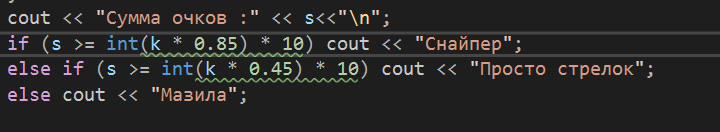


Рисунок 14 - Оценивание результата программой

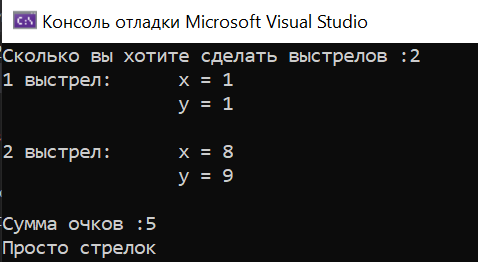


Рисунок 15 – Результат работы программы

**Вывод:** изучены методы реализации управляющих операторов, циклов, создана программа по расчёту суммы чисел на заданном интервале и программа, имитирующая стрельбу по мишени.