Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 РЕФЕРАТ

По дисциплине «Основы алгоритмизации и программирования»

На тему «Почему не стоит использовать оператор goto»

Выполнила:

Студентка1 курса 7 группы

Гриценко Анна Александровна

2023, Минск

Содержание

[Введение 3](#_Toc151080131)

[**Оператор goto** 3](#_Toc151080132)

[**Почему не следует использовать оператор goto?** 4](#_Toc151080133)

[**Заключение** 6](#_Toc151080134)

[**Список литературы** 7](#_Toc151080135)

# **Введение**

**Операторы**-это выражения, которые заканчиваются точкой с запятой. Программа в любом языке программирования состоит из большого количества различных операторов. В языке С различают пустые операторы, операторы выражений, составные операторы, условные операторы и другие, которые используются для решения различных задач. Все операторы обязательно заканчиваются точкой с запятой! Но для составных операторов, которые представляют собой набор последовательных логических операторов, точка с запятой не требуется.

Примеры различных операторов:

1. Операторы выражений: Number= c + c1;
2. Пустой оператор: ;
3. Условный оператор:

If(z<13){

Sum= sum + 1;

}

else if(z>13){

Sum2= sum2 + 1;

}

## **Оператор goto**

**Оператор goto-** это оператор безусловногоперехода, который позволяет изменить последовательность выполения операторов программы и перейти к определенному месту в коде.

Оператор goto имеет следующий вид:

goto Number;

Number- это метка, которая показывает , в какое место кода делается переход. Эту метку не требуется описывать. Важно отметить, что имя метки не может совпадать с названиями переменных и других элементов кода.

То есть с помощью этого оператора мы можем пропустить следующий оператор по порядку и перейти к другому.

Он также позволяет переходить как по ходу выполнения программы, так и в обратном направлении.

Оператор goto и его метка должны находится в одной и той же функции!

### **Почему не следует использовать оператор goto?**

Сравним два решения одной задачи, используя оператор goto и без него.

#include <iostream>

void main()

{

setlocale(LC\_CTYPE, "Russian");

double s, y, x = 9.e-3, r = 7.1, p;//вводим переменные

std::cout << "Введите p= ";

std:: cin >> p;

s = r + (5 \* x + p);//считаем s по формуле

s < 1.5;

goto a;//если выполняется предыдущее условие, то переходим к а

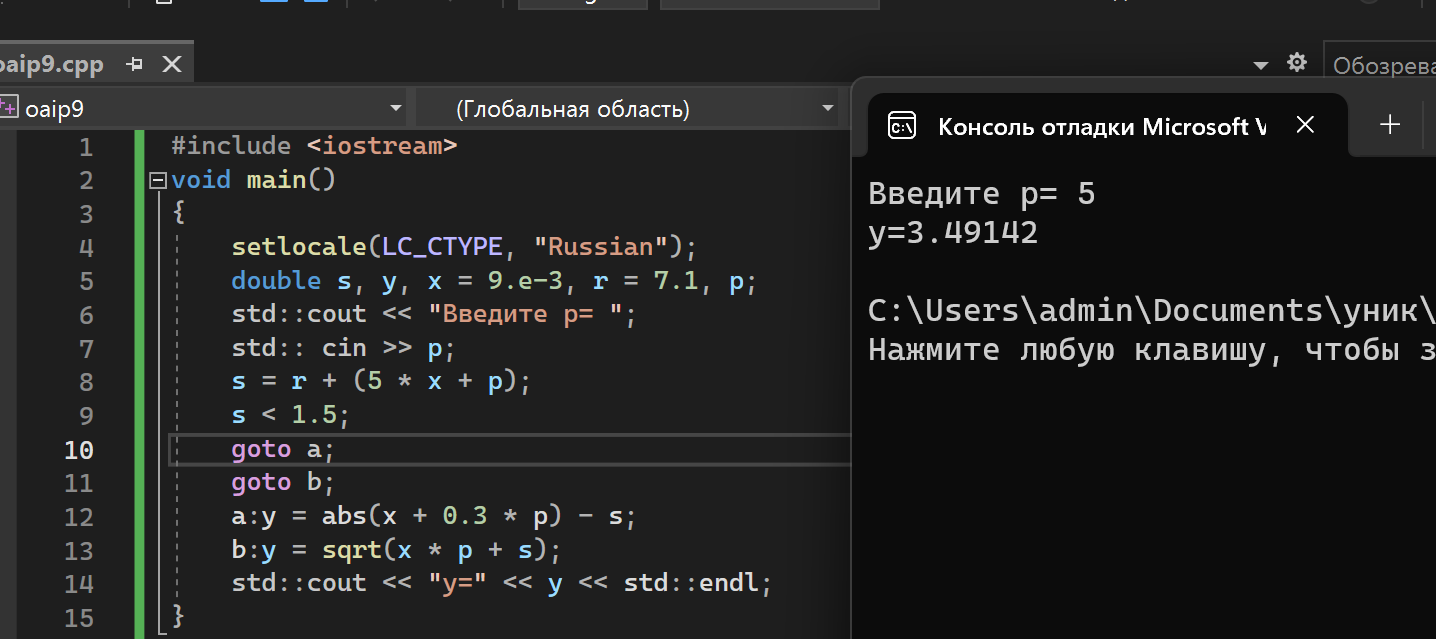
goto b;//если не выполняется условие, то переходим к b

a:y = abs(x + 0.3 \* p) - s;//находим y, если выполняется условие s < 1.5

b:y = sqrt(x \* p + s);//находим y, если не выполняется условие s < 1.5

std::cout << "y=" << y << std::endl;//выводим y

}



В примере 1 задача решена с помощью оператора безусловного ввода goto.

Пример 2:

#include <iostream>

void main()

{

setlocale(LC\_CTYPE, "Russian");

double s, y, x = 9.e-3, r = 7.1, p;//вводим переменны

std::cout << "Введите p= ";

std::cin >> p;

s = r + (5 \* x + p);

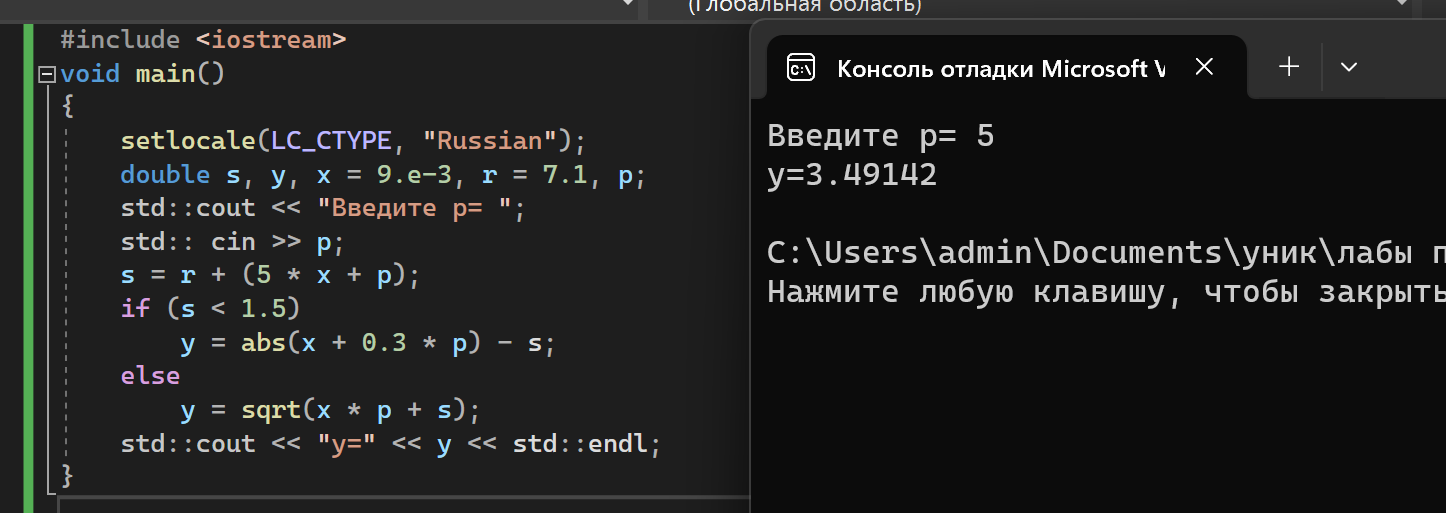
if (s < 1.5)

y = abs(x + 0.3 \* p) - s;//если выполняется условие s < 1.5

else

y = sqrt(x \* p + s);//если не выполняется условие s < 1.5

std::cout << "y=" << y << std::endl;

}

В примере 2 задача решена с помощью структуры if-else: If(задает условие, при котором некоторая операция будет выполнятся) и else(выполняет иною операцию, если условие для if не выполняется).

При использовании оператора goto код выглядит более запутанно и непонятно, а если пользоваться условным оператором if, код выглядит более понятно и наглядно видно какая оперция для какого условия выполняется. Становится очень сложно понять логику выполнения программы с оператором goto и следовать порядку ее выполнения.

Этот оператор используется в старых языках программирования (например, Basic). Но для языка С++ более эффективным решением будут написать программу с использованием условных операторов, функций и циклов.

Также очень сложно и достаточно долго найти ошибку в коде, если ты пишешь программу с goto. При большом количестве таких операторов будет очень много переходов в различные места кода и запутаться будет достаточно сложно.

#### **Заключение**

Оператор goto-оператор, кторый позволяет мгновенно переместиться в любую точку кода. Его называют оператором безусловного перехеда, так как он не требует никакого условия , чтобы перейти в то или иное место кода.Он многократно используется в старых языках программирования, но чаще всего его использование для языка С++ нецелесообразно. Оператор goto усложняет логику понимания кода и загромождает его. Поэтому лучше использовать условные операторы и циклы для написания разветвляющихся программ.

Оператор goto лучше использовать очень редко в С++, например в случаях, когда нужно выйти из нескольких вложенных циклах.

##### **Список литературы**

1. <https://bstudy.net/924013/tehnika/operator_bezuslovnogo_perehoda_pustoy_operator>
2. <https://victor192007.narod.ru/files/cpp05.html>
3. <https://ravesli.com/urok-66-operator-goto/>