## Лабораторная работа 2\*

## Логический тип данных

**1.1.** Вычислить значение выражений.

not odd (n) при n = 0

a = 10

b = 20

c = True

d = False

not odd(n) при n = 0

Выражения:

1. a + b

2. a - b

3. a \* b

4. a / b

5. a // b

6. a % b

7. a \*\* b

8. a > b

9. a < b

10. a >= b

11. a <= b

12. a == b

13. a != b

14. c and d

15. c or d

16. not c

17. not d

18. not odd(n)

Вычисления:

1. a + b = 10 + 20 = 30

2. a - b = 10 - 20 = -10

3. a \* b = 10 \* 20 = 200

4. a / b = 10 / 20 = 0.5

5. a // b = 10 // 20 = 0

6. a % b = 10 % 20 = 10

7. a \*\* b = 10 \*\* 20 = 100000000000000000000

8. a > b = 10 > 20 = False

9. a < b = 10 < 20 = True

10. a >= b = 10 >= 20 = False

11. a <= b = 10 <= 20 = True

12. a == b = 10 == 20 = False

13. a != b = 10 != 20 = True

14. c and d = True and False = False

15. c or d = True or False = True

16. not c = not True = False

17. not d = not False = True

18. not odd(n) = not odd(0) = True (так как 0 не является нечетным числом)

Таким образом, при указанных значениях переменных вычисленные значения выражений будут:

1. 30

2. -10

3. 200

4. 0.5

5. 0

6. 10

7. 100000000000000000000

8. False

9. True

10. False

11. True

12. False

13. True

14. False

15. True

16. False

17. True

18. True

**1.2**. Вычислить значения выражений:

a and b>a or b при a=false, b=true;

1. a and b

- a равно False, поэтому весь результат будет False.

2. b > a

- b равно True, а a равно False, поэтому b > a будет True.

3. a and b > a or b

- Имеем False and True or True.

- Сначала выполняется операция and, которая дает False.

- Затем выполняется операция or, которая дает True.

Таким образом, значение выражения a and b > a or b при a = False и b = True будет True.

**1.3.** Указать порядок выполнения операций при вычислении выражения (надписав над знаком операции номер действия):

1) (x>=0) or t and z or (y\*y<>4)

Порядок выполнения операций в этом выражении будет следующим:

1. (x>=0) - сравнение

2. t and z - логическая операция AND

3. (y\*y<>4) - сравнение

4. (x>=0) or t and z - логическая операция OR

5. (x>=0) or t and z or (y\*y<>4) - логическая операция OR

Таким образом, порядок выполнения будет следующим:

(x>=0) 1 or t 2 and z 2 or (y\*y<>4) 3

Сначала будут выполнены сравнения и логические операции and, а затем - оставшиеся логические операции or.

**1.4.** Объяснить ошибки в следующих записях:

not 2=5

1. Оператор not применяется к логическому выражению, а не к простому сравнению чисел.

2. Сравнение 2=5 является логическим выражением, которое возвращает значение False, так как 2 не равно 5.

3. Применяя оператор not к False, получаем True, так как not False эквивалентно True.

Поэтому, правильное выражение будет:

not (2 == 5)

Теперь давайте рассмотрим другие возможные ошибки в записи этого выражения:

1. not 2 = 5

- Здесь ошибка в том, что оператор сравнения = должен быть записан между двумя операндами, а не после оператора not.

- Правильная запись: not (2 = 5) или not (2 == 5).

2. not 2==5

- Здесь ошибки нет, это корректная запись выражения, эквивалентная not (2 == 5).

3. not(2=5)

- Здесь ошибки также нет, это корректная запись выражения, эквивалентная not (2 == 5).

Таким образом, основная ошибка в записи not 2=5 заключается в том, что оператор not должен применяться к логическому выражению, а не к простому сравнению чисел. Правильная запись этого выражения - not (2 == 5).

**1.5.** Записать на Паскале выражение, истинное при выполнении указанного условия и ложное в противном случае:

a, b, c — стороны треугольника. Записать, что треугольник существует

Для записи выражения, истинного при выполнении условия, что треугольник с сторонами a, b, c существует, на языке Паскаль, можно использовать следующий код:

(a + b > c) and (a + c > b) and (b + c > a)

Объяснение:

Для того, чтобы треугольник существовал, должны выполняться следующие условия:

1. Сумма длин любых двух сторон треугольника должна быть больше длины третьей стороны.

2. Поэтому мы проверяем три таких условия:

- a + b > c

- a + c > b

- b + c > a

Если все три условия истинны, то выражение (a + b > c) and (a + c > b) and (b + c > a) будет истинным, и можно сказать, что треугольник с сторонами a, b, c существует.

В противном случае, если хотя бы одно из этих условий ложно, то все выражение будет ложным, и треугольник с данными сторонами не может существовать.

Этот код можно использовать в программе на Паскале, например, для проверки возможности существования треугольника с заданными сторонами.

**1.6.** Составить программу, печатающую значение true, если указанное высказывание является истинным, и false в противном случае:

цифры данного трехзначного числа N являются членами арифметической прогрессии;

program CheckArithmeticProgression;

var

N: integer;

a, b, c: integer;

d1, d2: integer;

isArithmetic: boolean;

begin

writeln('Enter a three-digit number: ');

readln(N);

// Извлечение цифр числа

a := N div 100;

b := (N div 10) mod 10;

c := N mod 10;

// Проверка разности между первой и второй, и второй и третьей цифрами

d1 := b - a;

d2 := c - b;

// Проверка, являются ли цифры членами арифметической прогрессии

isArithmetic := (d1 = d2);

// Вывод результата

if isArithmetic then

writeln('true')

else

writeln('false');

end.

**1. 7.**  Переменной T присвоить значение выражения, истинного при выполнении указанного условия и ложного в противном случае:

не существует треугольника с длинами сторон a, b, c.

program CheckTriangleExistence;

var

a, b, c: integer;

T: boolean;

begin

// Ввод длин сторон треугольника

writeln('Enter the lengths of the triangle sides:');

write('a = '); readln(a);

write('b = '); readln(b);

write('c = '); readln(c);

// Проверка условия существования треугольника

T := NOT ((a + b > c) AND (a + c > b) AND (b + c > a));

// Вывод результата

if T then

writeln('The triangle does not exist.')

else

writeln('The triangle exists.');

end.

## Оператор выбора

* 1. Написать программу, которая по данному натуральному числу от 1 до 12 (номеру месяца) выдает все приходящиеся на этот месяц праздничные дни (например, если введено число 1, то: 1 января – Новый год, 7 января –Рождество).

program PrintHolidaysForMonth;

var

month: integer;

begin

// Ввод номера месяца

write('Enter the month number (1-12): ');

readln(month);

// Вывод праздников в зависимости от месяца

case month of

1:

begin

writeln('1 January - New Year''s Day');

writeln('7 January - Christmas');

end;

2:

begin

writeln('14 February - Valentine''s Day');

end;

3:

begin

// Нет праздников в марте

end;

4:

begin

writeln('1 April - April Fools'' Day');

end;

5:

begin

writeln('1 May - Labour Day');

end;

6:

begin

// Нет праздников в июне

end;

7:

begin

// Нет праздников в июле

end;

8:

begin

// Нет праздников в августе

end;

9:

begin

writeln('1 September - Knowledge Day');

end;

10:

begin

writeln('14 October - Defender of the Fatherland Day');

end;

11:

begin

writeln('4 November - National Unity Day');

end;

12:

begin

writeln('31 December - New Year''s Eve');

end;

else

writeln('Invalid month number.');

end;

end.