

# Level Up! Delphi

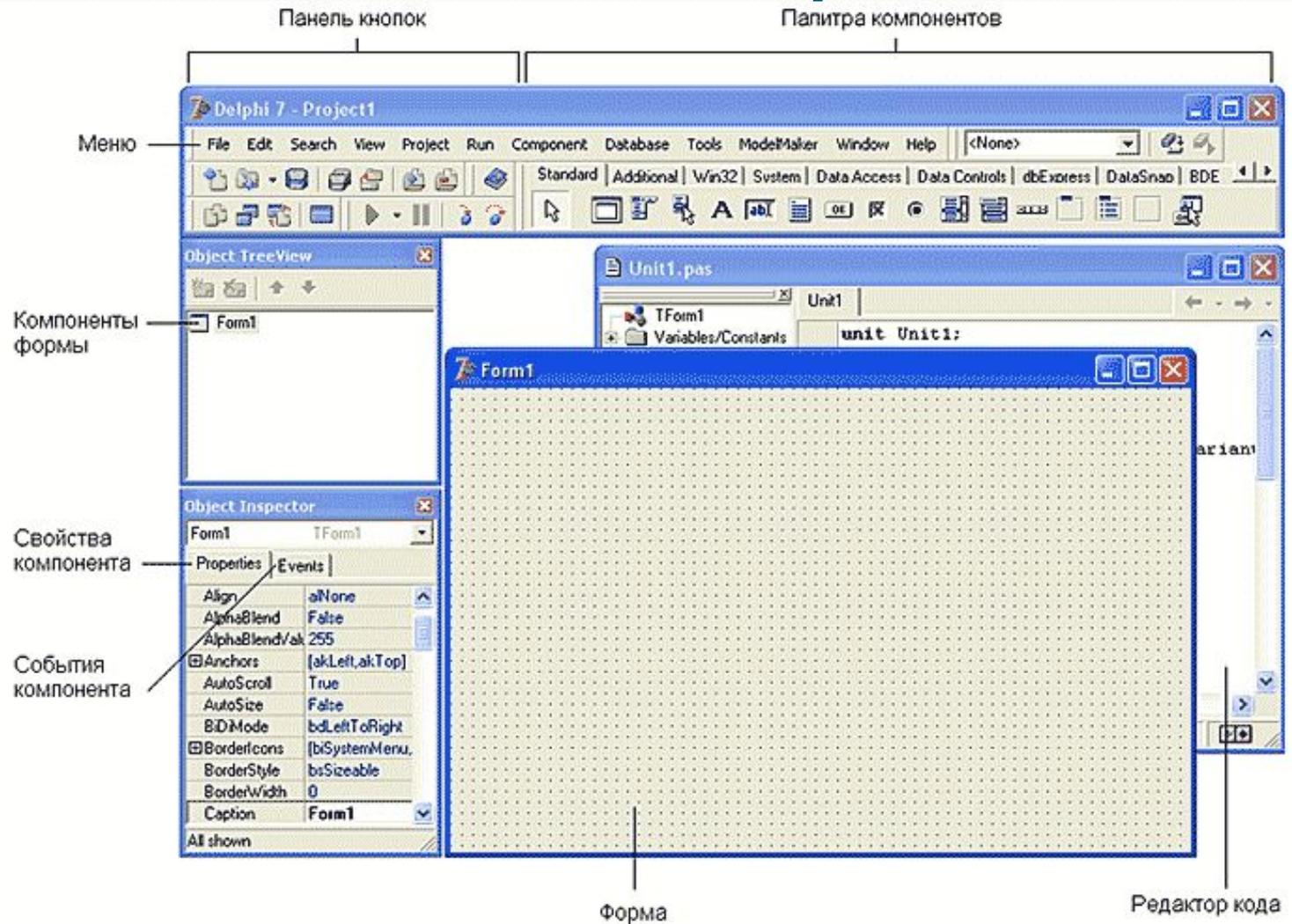
```
Writeln('Hello, world!');
```

**Королько Кирилл**  
**[vk.com/kirillkorolko](https://vk.com/kirillkorolko)**

**Level Up! (Delphi)**  
**[vk.com/levelupdelphi](https://vk.com/levelupdelphi)**

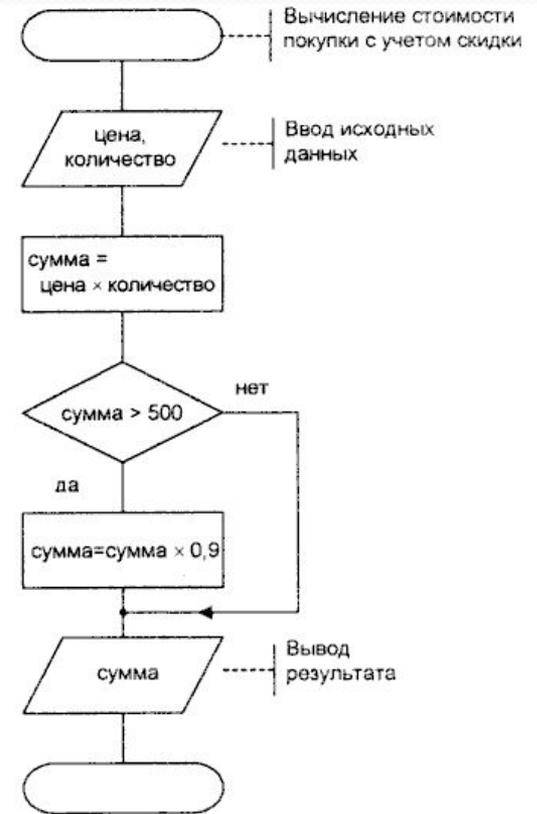


# Borland Delphi 7

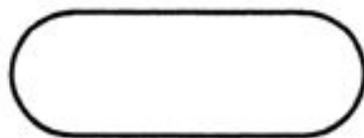


# Алгоритм

Алгоритм – это последовательность действий, которые необходимо выполнить, чтобы решить поставленную задачу.



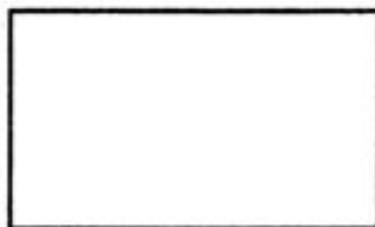
# При изображении алгоритма в виде блок-схемы используются специальные СИМВОЛЫ:



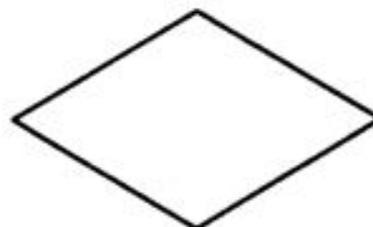
Начало/Конец



Ввод/Вывод



Обработка



Выбор

# Целые числа

Формат	Диапазон
Shortint	-128.. 127
Integer	-32 768.. 32 767
Longint	-2 147 483 648.. 2 147 483 647
Int64	$-2^{63}.. 2^{63} - 1$
Word	0..65535
Byte	0..255

# Числа с плавающей точкой

Формат	Диапазон	Кол-во значащих цифр
Real	$2.9 \cdot 10^{-39} - 1.7 \cdot 10^{38}$	11-12
Single	$1.5 \cdot 10^{-45} - 3.4 \cdot 10^{38}$	7-8
Double	$5.0 \cdot 10^{-324} - 1.7 \cdot 10^{308}$	15-16
Extended	$3.6 \cdot 10^{-4951} - 1.1 \cdot 10^{4932}$	19-20

# СИМВОЛЬНЫЙ ТИП

Язык Delphi поддерживает два символьных типа:  
AnsiChar и WideChar:

- тип **AnsiChar** — это символы в кодировке ANSI, которым соответствуют числа в диапазоне от 0 до 255;
- тип **WideChar** — это символы в кодировке Unicode, им соответствуют числа от 0 до 65 535.

# СИМВОЛЬНЫЙ ТИП

```
Var FirstSymbol, SecondSymbol, ThirdSymbol: Char;
```

```
...
```

```
FirstSymbol := 'a';
```

```
SecondSymbol := '5';
```

```
ThirdSymbol := '@';
```

# Строковый тип

Язык Delphi поддерживает три строковых типа:  
**ShortString, LongString, WideString:**

- тип **ShortString** представляет собой строки длиной от 0 до 255 СИМВОЛОВ;
- тип **LongString** представляет собой динамически размещаемые в памяти строки, длина которых ограничена только объемом свободной памяти;
- тип **WideString** представляет собой динамически размещаемые в памяти строки, длина которых ограничена только объемом свободной памяти. Каждый символ строки типа WideString является Unicode-символом.

В языке Delphi для обозначения строкового типа допускается использование идентификатора **String**. Тип String эквивалентен типу ShortString.

# Строковый тип

```
Var MyFirstString: String;  
    MySecondString: String[200];  
...  
MyFirstString := 'Hello,';  
MySecondString := 'world!';  
WriteLn(MyFirstString, ' ', MySecondString);
```

# Логический тип

Логическая величина может принимать одно из двух значений **True** (истина) или **False** (ложь). В языке Delphi логические величины относят к типу **Boolean**.

# Логический тип

```
Var Flag: Boolean;
```

```
    I: Integer;
```

```
...
```

```
Flag := True;
```

```
If (Flag) then
```

```
    I := 5
```

```
else
```

```
    I := 4;
```

```
WriteLn(I);
```

# Переменная

**Переменная** — это область памяти, в которой находятся данные, которыми оперирует программа. Когда программа манипулирует с данными, она, фактически, оперирует содержимым ячеек памяти, т. е. переменными.

Var A: Real; V: Real; I: Integer;

# Константы

**Константа** — это целое или дробное число, строка символов или отдельный символ, логическое значение.

В отличие от переменной, при объявлении константы тип явно не указывают. Тип константы определяется ее видом, например:

- 125 — константа целого типа;
- 0.0 — константа вещественного типа;
- ' выполнить ' — строковая константа;
- '\ ' — символьная константа.

# Константы

const

Bound = 10;

Title = 'Скорость бега';

pi = 3.1415926;

# Выражения

Оператор	Действие
+	Сложение
-	Вычитание
*	Умножение
/	Деление
div	Деление нацело
mod	Вычисление остатка от деления

# Стандартные математические функции

Функция	Значение
$Abs(n)$	Абсолютное значение $n$
$Sqrt(n)$	Квадратный корень из $n$
$Sqr(n)$	Квадрат $n$
$Sin(n)$	Синус $n$
$Cos(n)$	Косинус $n$
$Arctan(n)$	Арктангенс $n$
$Exp(n)$	Экспонента $n$
$Ln(n)$	Натуральный логарифм $n$
$Random(n)$	Случайное целое число в диапазоне от 0 до $n - 1$

# Условный оператор If

If (условие) then (операции 1)  
else (операции 2);

# Условный оператор If

```
If (a < b) then a := b  
    else b := a;
```

```
WriteLn(a);
```

```
If (a<b) then  
    begin  
        b := a;  
    end  
else WriteLn(b);
```

# Оператор условного выбора

## Case

**Case** (выражение) **of**

Константа 1: (операции 1) ;

Константа 2: (операции 2) ;

...

Константа n: (операции n) ;

**else** (операции n + 1);

**end;**

# Оператор условного выбора

## Case

```
Var i, b, k: integer;
```

```
...
```

```
ReadLn(i);
```

```
Case i of
```

```
1: b := 5;
```

```
2: b := 9;
```

```
3: begin
```

```
    b := 1;
```

```
    k := b + i;
```

```
end;
```

```
else b := i;
```

```
end;
```

# Домашняя работа №1

1. Найти площадь равностороннего треугольника со стороной  $a = 10$ .
2. Дана точка  $A$  с координатами  $(x, y)$ . Найти в какой из координатных четвертей она расположена.
3. Мальчик Вася каждый день ездит на метро. Утром он едет в школу, а вечером того же дня, обратно из школы, домой. Для того, чтобы немного сэкономить, он покупает электронную смарт-карту на  $X$  поездок. Когда он хочет зайти в метро, он прикладывает карту к турникету. Если на карте осталось ненулевое количество поездок, то турникет пропускает Васю и списывает с карты одну поездку. Если же на карте не осталось поездок, то турникет не пропускает Васю, и он (Вася) вынужден купить на этой же станции новую карту на  $X$  поездок и вновь пройти через турникет.

Вася заметил, что в связи с тем, что утром метро переполнено, покупать новую карту утром накладно по времени, и он может опоздать в школу. В связи с этим он хочет понять: будет ли такой день, что с утра, поехав в школу, окажется, что у него на карточке ноль поездок.

Вася больше никуда на метро не ездит и поэтому заходит в метро только на станции около дома и на станции около школы.

Спасибо за внимание 😊