

LAZARUS



**Типи даних. Змінні.
Операції. Вирази.
Стандартні
функції. Оператори.**

Технології, 10 клас

АЛГОРИТМ І ПРОГРАМА

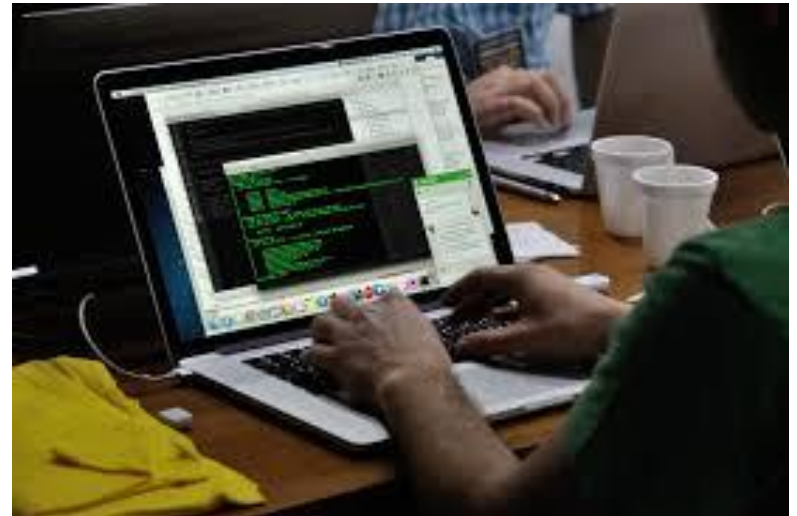
Комп'ютер – пристрій, створений людиною і призначений для виконання команд, написаних людиною.

Послідовність команд, які потрібно виконати у певному порядку, називається **алгоритмом**.

Алгоритм, складений для комп'ютера і записаний мовою, яку він розуміє, називається **програмою**.

Програми для комп'ютера створює програміст.

```
252 document.getElementById( 'bigimageDesc' ).innerHTML = description;
253
254
255 function updatePhotoDescription() {
256   if (descriptions.length > (page * 9) + (currentImage.substring(0, 1)))
257     document.getElementById( 'bigimageDesc' ).innerHTML = description;
258 }
259
260
261 function updateAllImages() {
262   var i = 1;
263   while (i < 10) {
264     var elementId = 'foto' + i;
265     var elementIdBig = 'bigimage' + i;
266     if (page * 9 + i - 1 < photos.length) {
267       document.getElementById( elementId ).src = 'images/' + photos[
268
```



МОВА ПРОГРАМУВАННЯ

Штучні мови, створені людиною для написання команд для комп'ютера, називаються **мовами програмування**. Іншими словами, програми для комп'ютера пишуться на певній мові програмування.

З часу створення перших комп'ютерів було створено понад **дві з половиною тисячі** мов програмування. Одні з них відійшли у минуле, інші використовуються і зараз. Професійні програмісти зазвичай застосовують в своїй роботі лише декілька мов програмування.



МОВА ПРОГРАМУВАННЯ

Мови програмування поділяються на **мови низького рівня** (близькі до команд, які може виконувати процесор комп'ютера – машинних кодів) та **мови високого рівня** (близькі до мови людини). Середовище LAZARUS використовує мову високого рівня Object Pascal.



МОВА ПРОГРАМУВАННЯ

Програми, написані мовами високого рівня, потрібно потім перевести (транслювати) у машинні коди, адже лише такі програми може виконати процесор комп'ютера. Цим перекладом займаються спеціальні програми, котрі називають **трансляторами**. Транслятори поділяються на інтерпретатори і компілятори. Object Pascal використовує компілятор.



СИНТАКСИС І СЕМАНТИКА

Кожна мова програмування має три складові – алфавіт, синтаксис і семантику.

Алфавіт – це символи, що використовуються для запису програм даною мовою (букви, цифри, спеціальні символи, ключові слова).

Синтаксис – це правила запису із символів алфавіту конструкцій мови (правила створення синтаксично правильних команд, які середовище програмування зможе потім перевести у машинні коди для виконання процесором).

Семантика – правила надання практичного сенсу синтаксично правильним мовним конструкціям (адже синтаксично правильно записані команди можуть при їх виконанні не привести до жодного результату замість бажаного). Справді, з правильно записаних слів можна скласти як літературний твір-шедевр, так і безглуздий набір

ЕЛЕМЕНТИ ПРОГРАМИ

Елементи програми - це мінімальні неподільні її частини, що несуть у собі певну значимість. До елементів відносяться:

- зарезервовані слова;
- ідентифікатори;
- типи;
- константи;
- змінні;
- мітки;
- підпрограми;
- коментарі.

```
and          except      label       resourcestring
array        exports    library     set
as           file        mod         shl
asm          finalization nil         shr
begin        finally     not         string
case         for         object      then
class        function    of          threadvar
const        goto        or          to
constructor  if          out         try
destructor   implementation packed     type
dispinterface in          procedure   unit
div          inherited   program     until
do           initialization property    uses
downto       inline      raise       var
else         interface   record      while
end          is          repeat      with
                                xor
```

Зарезервовані слова – це англійські слова, що означають для середовища необхідність виконання певної дії, елементи мовних конструкцій; виділяються іншим кольором.

ТИПИ ДАНИХ

До основних типів даних мови Object Pascal відносяться:

- **цілі числа** (універсальний тип – integer)
- **дробові числа** (універсальний тип - real)
- **символи** (char) – поодинокі символи, наприклад ‘Z’, ‘Ж’
- **рядки** (string) – слова, фрази, наприклад ‘Мені 16 років’
- **логічний тип** (boolean) – приймає значення лише **так** (true - істина) чи **ні** (false - хиба).

Цілі типи

Тип	Діапазон	Формат
Shortint	-128..127	8 бітів
Smallint	-32 768 .. 32 767	16 бітів
Longint	-2 147 483 648 .. 2 147 483 647	32 біти
Int64	$-2^{63} .. 2^{63} - 1$	64 біти
Byte	0..255	8 бітів, без знаковий
Word	0..65 535	16 бітів, без знаковий
Longword	0 .. 4 294 967 295	32 біти, без знаковий

Дійсні (дробові) типи

Тип	Діапазон	Значущі цифри	Байтів
Real	$2.9 * E^{-39} .. 1.7 * E^{38}$	11-12	06
Single	$1.5 * E^{-45} .. 3.4 * E^{38}$	7-8	04
Double	$5.0 * E^{-324} .. 1.7 * E^{308}$	15-16	08
Extended	$3.4 * E^{-4932} .. 1.1 * E^{4932}$	19-20	10
Comp	$1.0 .. 9.2 * E^{18}$	19-20	08
Currency	$0.0001 .. 9.2 * E^{14}$	19-20	08

Integer еквівалентний LongInt

ЗМІННІ

Змінна — це область пам'яті комп'ютера, в якій знаходяться дані, котрими оперує програма. Кожна змінна повинна мати своє неповторне і зрозуміле ім'я, яке придумує програміст. Як ім'я змінної можна використовувати послідовність з букв латинського алфавіту, цифр і деяких спеціальних символів, однак **першим символом в імені змінної повинна бути лише буква** (наприклад, *suma*, *name1*, *my_age* тощо). Пропуск (пробіл) в імені змінної використовувати не можна.

Object Pascal не розрізняє великі (прописні) і малі (рядкові) букви в іменах змінних: імена *SUMA*, *Suma* і *suma* - це одна і та ж змінна.

Змінні, що будуть використані у програмі, обов'язково оголошуються в розділі оголошення змінних, який починається словом **var** (variable - змінна). При оголошенні змінних вказується їх ім'я

При

```
var
  Form1: TForm1; z1, z2, z3: longint; my_name: string;
```

КОНСТАНТИ

В мові Object Pascal існує два види констант: звичайні і іменовані.

- **звичайна константа** — це **ціле** або **дробове число**, **рядок символів** або **окремий символ**, **логічне значення**.
- **іменована константа** — це ім'я (ідентифікатор), яке в програмі використовується замість самої константи.

Розділювачем у десяткових дробах є крапка (наприклад, **9.8**). Стандартний вид числа **$2,34 \cdot 10^{-12}$** записують так: **2.34e-12** (буква E може бути малою чи великою). Окремі символи чи рядки обмежують апострофом (**'а'**, **'мама'**). При наявності у рядку апострофа його подвоюють: **'шкіряний м'яч'**. Логічні значення — **true**, **false**.

Іменована константа, як і змінна, перед використанням повинна бути оголошена в розділі оголошення констант, який починається словом **const**. Приклад оголошення констант:

```
const
Bound = 10;
Title = 'Швидкість автомобіля';
pi = 3.1415926;
```

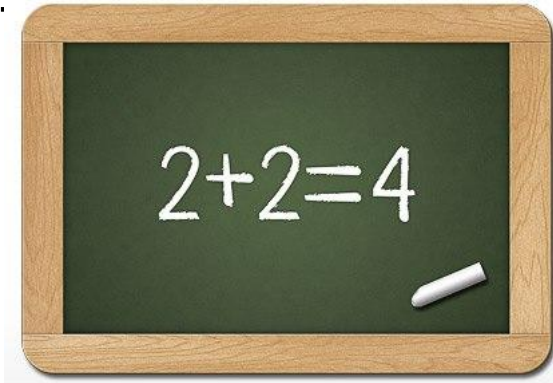


ВИРАЗИ

З допомогою констант і змінних та знаків операцій і дужок можна створювати вирази. Зрозуміло, що значення виразу має певний тип даних (один із стандартних).

Приклади виразів:

- $12+3.4$ (матиме тип *real*)
- $x1>2$ (матиме тип *boolean*)
- `My_name + ' - student'` (матиме тип *string*).



В **арифметичних виразах** використовуються операції додавання (+), віднімання (-), множення (*) та ділення (/). Можливе також використання операцій *div*, *mod* (наприклад, $x + z1 \bmod 4 - z2 \operatorname{div} 10$). При використанні різних типів результат матиме тип з більш ширшим діапазоном значень.

Логічні вирази використовують як операції порівняння ($>$, $<$, $=$, $>=$, $<=$, $<>$), так і логічні операції *not*, *and*, *or*, *xor*, наприклад:

- $(x >= -5) \text{ and } (x < 16)$
- $((z1 < -10) \text{ or } (z1 > 15)) \text{ and } (z2 >= -20)$

ОПЕРАНДИ, ОПЕРАТОРИ

Вираз складається з **операндів** і **операторів**. Оператори знаходяться між операндами і позначають дії, які виконуються над операндами. У виразі **12 + 6** числа 12 і 6 – операнди, дія **+** (додавання) є оператором.

Як операнди виразу можна використати змінну, константу, функцію або інший вираз. Основні оператори алгебри приведені в

Оператор	Дія	Оператор	Дія
+	Додавання	/	Ділення
-	Віднімання	DIV	Ціла частина від ділення
*	Множення	MOD	Остача від ділення

При записі виразів між операндом і оператором, за винятком операторів DIV і MOD, пропуск можна не ставити.

При обчисленні значень виразів слід враховувати, що оператори мають різний пріоритет. Так у операторів *****, **/**, **DIV**, **MOD** більш високий пріоритет, ніж у операторів **+** і **-**. Пріоритет дужок ще вищий. Число відкриваючих дужок (має бути рівний числу закриваючих дужок).

СТАНДАРТНІ ФУНКЦІЇ

Object Pascal пропонує великий набір математичних та інших функцій для дій над числами, датами, рядками тощо (особливо при використанні додаткових модулів типу *math*).

Стандартні математичні функції:

Функція	Значення
<i>Abs (n)</i>	Абсолютне значення (модуль) числа n
<i>Sqrt (n)</i>	Квадратний корінь з невід'ємного числа n
<i>Sqr (n)</i>	Квадрат числа n
<i>Sin (n)</i>	Синус кута n (кут – в радіанах)
<i>Cos (n)</i>	Косинус кута n (кут – в радіанах)
<i>Arctan (n)</i>	Арктангенс числа n
<i>Exp (n)</i>	Експонента числа n
<i>Ln (n)</i>	Натуральний логарифм числа n
<i>Random (n)</i>	Випадкове ціле число в діапазоні від 0 до $n-1$

СТАНДАРТНІ ФУНКЦІЇ

Функції для перетворення величин одних типів у інші типи:

Функція	Значення функції
Chr(n)	Символ, код якого рівний n
IntToStr(k)	Рядок, що є зображенням цілого числа (перетворення цілого числа у рядок, наприклад 123 □ '123'), <i>integer to string</i>
FloatToStr(n)	Рядок, що є зображенням дійсного числа n (перетворення дійсного числа у рядок, наприклад, 1.23 □ '1.23')
FloatToStrF(n, f, k, m)	Перетворення дійсного числа у рядок з потрібним форматом. Указують: f — формат (спосіб зображення); k — потрібна точність (загальна кількість цифр); m — кількість цифр після крапки
StrToInt(s)	Перетворення рядка s у ціле число, якщо це можливо, наприклад '123' □ 123 ; <i>string to integer</i>
StrToFloat(s)	Перетворення рядка у дійсне число, якщо це можливо, наприклад '1.23' □ 1.23
Round(n)	Ціле число, отримане шляхом округлення n за відомими правилами
Trunc(n)	Ціле число, отримане шляхом відкидання дробової частини дійсного числа n
Frac(n)	Дробове число, що є дробовою частиною дійсного числа n

ОПЕРАТОР ПРИСВОЄННЯ

Основною конструкцією Object Pascal, так само як і багатьох інших мов програмування, є **оператор**. Оператор означає деяку дію, яке необхідно виконати. Для запису операторів використовують вказані раніше **ключові слова**. Оператори розділяються крапкою з комою. Загальноприйнято записувати окремі оператори окремими рядками.

У кожного оператора Object Pascal є строго певний синтаксис-правила побудови, яких потрібно дотримуватись для уникнення помилок під час виконання програми.

Найпростіший і найвживаніший – **оператор присвоєння**, який присвоює змінній її значення. Синтаксис оператора присвоєння такий: **ім'я змінної := її значення**

Значенням змінної може бути константа, змінна чи вираз.

Приклади: A:=2.7; Z1:= abs(x)*int(a+3.5) ;

 N:=N+1; k:=my_name + ' - girl';