

# Мови програмування



- Асемблер
- Синтаксичний цукор
- Імперативні мови програмування
- Функціональні мови програмування

# Асемблер

```
mov     esi, [esi]
PrintLoop:
mov     dl, [esi:si]
inc     si
mov     ah, 2
int     21h
cmp     si, BuffSize
jne     @NotAtEnd
xor     si, si
... + AtEnd:
si [esi:Head
mov     PrintLoop
```

- Кожен процесор має набір інструкцій
- Програма з цих інструкцій називається автокодом
- Асемблер – транслятор з автокоду на машинну мову.

# Синтаксичний цукор

Синтаксичний цукор — термін, що позначає доповнення синтаксису мови програмування, які не додають нових можливостей, а роблять використання мови більш зручним для людини.

# Приклади

В мові паскаль доступ до елементу масиву можна здійснювати кількома способами:

```
var
  a: array[0..9] of byte;
begin
  a[5] := 7;
  (pbyte)((integer)@a+5*sizeof(byte))^ := 7;
end;
```

Очевидно, що перший спосіб коротший і зручніший, але він все одно компілятором приведеться до другого.

Всі є аналогічні конструкції.

# Приклади

В Сі++ для збільшення значення змінної  $x$  на 1, можна:

```
x = x+1; // або  
x++; // або  
++x;
```

Але оператор “++” можна комбінувати з іншими операторами, код:

```
int i = 5;  
i = ++i + ++i;
```

призведе до (UB) невизначеної поведінки.

Значення  $i$  може бути 13 або 14 в залежності від компілятора.

# Критика синтаксичного цукру

Другий приклад показує що синтаксичний цукор може бути шкідливим. Він призводить до ускладнення програм і трансляторів, а також до помилок.

Програміст повинен використовувати тільки ясні конструкції, щоб уникнути помилок.

Найпослідовніший критик синтаксичного цукру – Ніклаус Вірт, він також є творцем мови Pascal, Oberon, та багатьох інших.

# Імперативні мови програмування

## Приклади

- C
- C++
- Oberon
- Java

Імперативне програмування — парадигма програмування, згідно з якою описується процес отримання результатів як послідовність інструкцій зміни стану програми. Поширений синонім імперативному програмуванню є процедурне програмування.

Програміст просто вказує що потрібно зробити і **як** це зробити.

# Мова програмування Сі



С (англ. C) — універсальна, процедурна, імперативна мова програмування загального призначення, розроблена у 1972 році Денісом Рітчі у Bell Telephone Laboratories з метою написання на ній операційної системи UNIX.

Імовірно, є найпопулярнішою у світі мовою програмування за кількістю вже написаного на ній програмного забезпечення та кількості програмістів, котрі її знають.

С — мінімалістська мова програмування.

С називають асемблером, незалежним від процесора і платформи.



# Мова програмування Сі++



Мова програмування С++ розроблена на основі С, і була отримана у наслідок додавання Б'ярном Страуструпом до неї об'єктно-орієнтованої функціональності із С-подібним синтаксисом.

При створенні С++ прагнули зберегти сумісність з мовою С. Більшість програм на С справно працюватимуть і з компілятором С++. С++ має синтаксис, заснований на синтаксисі С.

Найважливішим нововведенням є об'єктна модель, що дозволяє використовувати засоби об'єктно орієнтованого програмування.

В програмі на С++ можна визначати структури даних спільно з методами для їх обробки – класи.

# Мова програмування Oberon



Oberon - мова програмування високого рівня розроблена Ніклаусом Віртом, творцем мови Pascal.

Особливості:

- автоматичне повернення виділеної пам'яті системі;
- екстремально простий синтаксис;
- орієнтація на модульне програмування;
- відсутність синтаксичного цукру.

За твердженням Вірта, розробники мови Java використали його напрацювання в мові Oberon.

# Синтаксис Oberon

Щоб досягти максимальної надійності та продуктивності трансляції, було здійснено значне спрощення мови.

Опис його синтаксису вмістився на одній сторінці, повний опис мови має близько 20 сторінок, що вдвічі менше за опис Модули-2. Oberon є якщо не мінімальною, то, у будь-якому разі, однією з найменших за обсягом універсальних мов програмування високого рівня.

# Методи в Oberon

До мови не було включено синтаксичного поняття методів (процедур і функцій), що пов'язані з типом, тому що цей механізм у найзагальнішому вигляді можна легко змодельовувати, створюючи в записах поля процедурного типу й надаючи їм значення процедур, що відповідають методам.

## Pascal

```
type
  TCar = object
  private
    px, py: INTEGER;
  public
    procedure move(x, y: INTEGER);
  end;
```

## Oberon

```
TYPE
  Car = RECORD
    px: INTEGER;
    py: INTEGER;
    move: PROCEDURE(x, y: INTEGER);
  END;
```

# Мова програмування Java



Java – об'єктно-орієнтована мова програмування з Сі-подібним синтаксисом.

Основна особливість мови – компіляція не для конкретної платформи, а в байт-код. Цей код потім виконує віртуальна машина.

Такий підхід дозволяє обмежитися одним компілятором для мови Java.

На конкретній платформі для виконання програм має бути встановлена віртуальна машина (JVM).

JVM доступна для всіх основних сучасних платформ, тому про програми, що скомпільовані у Java байткод теоретично можна сказати «Написано один раз, працює скрізь».

# Функціональні мови програмування

## Приклади

- Haskell
- Hore
- F#
- LISP
- Scala

Функційне програмування — парадигма програмування, яка розглядає програму як обчислення математичних функцій та уникає станів та змінних даних. Функційне програмування наголошує на застосуванні функцій, на відміну від імперативного програмування, яке наголошує на змінах в стані та виконанні послідовностей команд.

# Функціональні мови програмування

Основна ідея імперативного програмування така: “Я скажу — як”. Кожне твердження в програмі є вказівкою того, що треба зробити на наступному кроці.

В функційних мовах програмування немає змінних, немає оператора присвоєння. Вся програма є функцією.

Програміст описує задачу. Не вказує як конкретно обчислювати.

