

# Типи даних у функціональному програмуванні

## Лекція 3



# Основа для творення типів даних Lisp

символьні вирази,  
S-вирази  
(Symbolic expresion)

# S-вирази

- це або атом, або список, або пара.

Атом – це найпростіший об'єкт, пара і список – складені дані.

Як правило, алфавіт функційної мови складається з усіх допустимих друкованих символів (латинські букви, цифри, знаки пунктуації, спеціальні символи тощо).

Різні діалекти можуть мати відмінності від класичної версії LISP.

Деякі символи мають спеціальне призначення

*() . пробіл*

Їх не можна використовувати без призначення

# Атом

Найпростіший об'єкт Lisp, з яких будуються всі інші дані.

Схематично організацію атомів в пам'яті можна представити у вигляді одиночної комірки.

A rectangular box with a black border, representing a memory cell for an atom.

atom

Атом розглядається як неділимий цілий об'єкт.  
Атоми бувають двох типів - символічні і числові.

# Символьні атоми

будуються з послідовності літер та/або цифр, при цьому повинен бути щонайменше один символ, відмінний від цифрового знаку.

Наприклад, 12LISP, scheme13, сі, 10A

До символьних атомів застосовується тільки одна операція: порівняння на тотожність.

# Логічні величини

Серед символічних атомів є два спеціальних, які позначають логічні значення істина (T) та неістина (NIL). У Scheme ці величини позначаються відповідно як `#t` та `#f`.

# Числові атоми

– це звичайні числа.

Наприклад, 124, -344, 4.5, 3.055E8.

Числа - це константи, над якими можна виконувати усі допустимі арифметичні операції та порівняння.

# Пара

Об'єднання двох елементів в одне ціле називається парою.

Елементами пари можуть бути об'єкти допустимої у функційному програмуванні структури, тобто атом, пара, список.

У деяких версіях ФМ пари окремо не розглядаються, а тільки як підвид списків.



Пара позначається в дужках, між елементами ставиться пробіл крапка, пробіл.

Наприклад, (fdg . hjk), (45 . Jk), ((2 . Df) . n)

Схематично пара представляється у вигляді комірки поділеної на дві частини


fdg	hjk
-----	-----

# Список

- це послідовність зв'язаних між собою елементів, кожен з яких є або атомом, або списком, або парою.

Списки позначаються в дужках, елементи списку розділяються пробілами.

Наприклад, (A D (G H) W) – список з чотирьох елементів: трьох атомів та списку з двох елементів



Члени списку організовані в пам'яті у вигляді послідовностей комірок, розділених на дві частини.


У першій частині комірки вказується інформація про член списку. Це може бути безпосереднє значення, якщо інформація подається атомом, або адреса комірки, де є перший елемент вкладеного об'єкту, в протилежному випадку.

У другій частині комірки вказується адреса зв'язку, тобто адреса знаходження наступного члена заданого списку.

Якщо член списку є останнім, то у другій частині комірки вказується символ NIL ("ніщо"), який позначається у вигляді діагоналі другої частини останньої комірки.



# Наприклад



Список - це багаторівнева або ієрархічна структура даних, в якій відкриваючі та закриваючі дужки знаходяться в строгій відповідності.

Список, в якому немає ні одного елемента, називається пустим списком і позначається як "()" або символом NIL. У такому випадку NIL - це і список, і атом одночасно.

Пустий список грає таку ж важливу роль в роботі зі списками, що і нуль в арифметиці. Пустий список може бути елементом інших списків.

# Вкладеність складених S-виразів

У парах та списках допускається вкладеність, тобто пари, списки можуть бути складовими інших пар чи списків. Якщо пара чи список складається лише з атомів, то кажуть, що вони мають нульовий рівень вкладеності. Якщо у списку (парі) у вигляді елемента містить список (пара), які складаються лише з атомів, то кажуть, що вони мають перший рівень вкладеності і т.д.

Допускається довільний рівень вкладеності.

# Зв'язок між парою та списком

Кожен список має голову і хвіст (2 частини).

Голова списку – це перший елемент списку, а хвіст – залишок списку без голови (також список).

Отже, кожен список може бути представлений у вигляді пари.

# dot – нотація списку


Отже, перший елемент – голова списку,  
А другий елемент пари – хвіст списку.

Наприклад, (12 34 56) – список, звичайна  
форма представлення.


(12 . (34 56))

(12 . (34 . (56 . ())))






Графічне представлення списків дає  
можливість у цьому переконатися  
наочно.




???

Чи кожна пара може бути представлена  
списком?



Список – форма представлення пасивних  
(просто набір даних) та активних даних  
(функція).

Список є рекурсивною структурою.



???

У чому полягає рекурсивність списку?