

## 1. Варіант № 7

### 1.1. Поясніть поняття мови Пролог «предикат». Розгляньте правила запису предикатів у Пролог-програмі. Проілюструйте прикладами.

Предикат — це засіб мови Пролог, який описує відношення між своїми аргументами і набуває значень «успіх» або «неуспіх» («істина» або «хибність»). Предикати записуються за такими правилами:

1. Ім'я кожного предикату повинно починатись з маленької літери.
2. Спочатку записується ім'я предиката, а потім, через кому і в дужках — об'єкти, яких він стосується.
3. Тіло предикатів повинно закінчуватись символом крапки — «. ».

Приклади предикатів:

```
human(john).  
father(david, john).  
abc(def, ghi, jkl, m).  
tree.  
p(X) :- r(X).  
p(X, _) :- r(X, _).  
p(value) :- a.  
p :- a, b.
```

Розглянемо предикат  $p(X)$ :

```
p(X) :- r(X).
```

Тут:

1.  $p, r$  — ім'я предикатів.
2.  $X$  — аргумент предикату (точніше, двох предикатів:  $p$  і  $r$ ).
3.  $:-$  — символ, що означає імплікацію.
4. Права частина  $r(X)$  . — тіло предикату.

Суть цього предикату така: предикат  $p(X)$  буде істинним, якщо предикат  $r(X)$  буде істинним.

**1.2. Поясніть, що називається звільненням змінної в мові Пролог. Наведіть програму — приклад використання звільнення змінної. Поясніть, яку роль відіграє звільнення змінної при виконанні цієї програми.**

Звільнення змінної в мові Пролог — це процес, при якому змінна повертається до неконкретизованого стану. Змінна називається неконкретизованою, якщо вона не пов'язана з жодним значенням. Звільнення змінної відбувається, коли Пролог шукає розв'язки поставленого запиту, і стається під час відкату (backtracking).

Наприклад, напишемо базу знань, що описує деяких тварин:

```
animal(bear).  
animal(dog).  
animal(cat).
```

Сформуємо запит `animal(X)`, який використовуватиме звільнення змінної. На початку виконання цього запиту змінна `X` буде неконкретизованою. Пролог починає шукати розв'язок запиту і пов'язує змінну `X` зі значенням `bear`. При такому значенні Пролог повідомляє, що предикат істинний, звільнює змінну і переходить до наступного факту.

В цій програмі роль звільнення змінної у тому, щоб знайти інші розв'язки, що задовольняють запит.

**1.3. Задача. Напишіть логічну програму, яка введе з клавіатури до списку елементи-числа, після чого проаналізує цей список і визначить найбільше з чисел, що до нього входять.**

```
% ISO-Prolog  
% Обгортка для предиката findmax_3 на 2 параметри  
findmax([H | T], Res) :-  
    findmax_3(T, H, Res).  
  
% Термінальна вітка: для пуского списку максимумом будуть передані  
% параметри  
findmax_3([], R, R).  
% Якщо поточний елемент більше поточного максимуму CurMax, змінити  
% поточний максимум CurMax на поточний елемент H і шукати максимум  
% у хвості списку  
findmax_3([H | T], CurMax, Res) :-  
    H > CurMax,
```

```
    findmax_3(T, H, Res).
% Інакше не змінювати максимум і шукати у хвості списку
findmax_3([H | T], CurMax, Res) :-
    H <= CurMax,
    findmax_3(T, CurMax, Res).

% Оболонка для єдиного параметра
list_from_stdin(R) :-
    list_from_stdin([], R).
% Зчитує дані, що вводить користувач. Елементи додаються у початок
% списку
list_from_stdin(InList, OutList) :-
    writeln("Please enter a number (or anything else to stop):"),
    read(Input),
    % Перевірити, чи є введені дані числом
    (
        integer(Input)
    ;   float(Input)
    ),
    % Додати елемент у початок списку
    list_from_stdin([Input | InList], OutList).
% Якщо введені дані не були числом, вивести поточний список.
list_from_stdin(L, L).

main :-
    list_from_stdin(L),
    write("Your list: "),
    writeln(L),
    findmax(L, Max),
    write("Max in your list: "),
    writeln(Max).
```