جامـــــعـة المـــــستـقبـل
AL MUSTAQBAL UNIVERSITY

كلية العلـوم

قسـم علوم الذكاء الاصطناعي

المحاضرة الاولى

Introduction to Prolog

المادة: Prolog languages

المرحلة:الاولى

اسم الاستاذ: م.م اية محمد حسين محمد علي

1. What is Prolog?

Prolog (programming in logic) is one of the most widely used programming languages in artificial intelligence research. It is used for solving problems that involve objects and relationships between objects.

Programming language families

Executive(procedural)	Declarative
BASIC, FORTRAN, C, Pascal, Java	LISP, Prolog, ML
Programmer tells the computer how to solve the problem.	Programmer tells the computer what problem should be solved.

2. Prolog Properties and Uses

The lecture highlights four Properties of Prolog: it can solve some problems with fewer lines of code; it is important for AI applications and expert systems; programs consist of facts and rules; and inference is used to search for all possible answers.

Less code A particular problem may be expressed with fewer code lines.	AI relevance Prolog is an important tool for AI applications and Expert Systems (ES).
Facts and rules A Prolog program consists of facts and rules used to solve the problem.	Inference The programmer provides knowledge; inference searches for possible answers.

Common uses listed in the lecture: grammar and language processing, knowledge representation and reasoning, unification, pattern matching, planning, and search.

Facts

like(john, cat).

like(john, flower).

like(ali, cat).

Queries

1) like(john, X).

X = cat

X = flower

2) like(X, cat).

X = john

X = ali

3) like(X, Y).

X = john , Y = cat

X = john , Y = flower

X = ali , Y = cat

Domains

s=symbol.
i=integer.

Predicates

age(s, i)

Clauses

age(a,10).
age(b,20).
age(c,30).

Goal

1.	age(a,X).	*Answer:* X=10	1 solution
2.	age(X,20).	*Answer:* X=b	1 solution
3.	age(X,Y).	*Answer:* X=a Y=10, X=b Y=20, X=c Y=30.	3 solutions
4.	age(_,X).	*Answer:* X=10, X=20, X=30.	3 solutions
5.	age (b,20).	*Answer:* Yes	
6.	age(_,_).	*Answer:* Yes	

***Note:** '_' means don't care

3. Terms, Constants, and Variables

The central data structure in Prolog is the term. The lecture names four kinds of terms: atoms, numbers, variables, and compound terms. Atoms and numbers are sometimes grouped together as atomic terms.

Constants and atoms
A constant may be an integer number, a real number, or an atom. Any name beginning with a lowercase letter is an atom, and many non-alphanumeric symbol sequences may also act as atoms.

Examples: school, g, +, *, -

Variables
A variable begins with an uppercase letter or an underscore, followed by letters, digits, or underscores.

Examples: X, Y, Variable, _tag, X_526, List, List24, _head, Tail, _input, Output

Prolog Terms

The central data structure in Prolog is that of a term. There are four kinds of terms: atoms, numbers, variables, and compound terms. Atoms and numbers are sometimes grouped together and called atomic terms.

Constants

Integer numbers, real numbers, or an atom. Any name that starts with a lowercase letter (followed by zero or more additional letters, digits, or underscores) is an atom. Atoms look like variables of other languages but are treated as constants in Prolog. Sequences of most non-alphanumeric characters (+, *, -, etc.) are also atoms.

Variables

A variable is any name beginning with an uppercase letter or an underscore, followed by zero or more additional letters, digits, or underscores.

For example:

X, Y, Variable, _tag, X_526, List, List24, _head, Tail, _input, and Output are all Prolog variables.

4. Data Types

The lecture lists five data types used to define program entries.

Type	Meaning	Examples
Integer	numerical value	1, 20, 0, -3, -50
Real	decimal value	2.4, 3.0, -2.67
Char	single character in single quotes	'a', 'C', '1'
String	sequence of characters in double quotes	"Computer", "134", "a"
Symbol	single character or sequence beginning with a small letter, without quotes	g, school

Note: The lecture examples emphasize formatting rules, such as single quotes for char values and double quotes for strings.

5. Prolog Program Structure

A Prolog program can contain five segments: domains, database, predicates, clauses, and goal. The lecture notes that not every program must contain all five, but predicates, clauses, and goal are required.

Domains	Database	Predicates	Clauses	Goal
defines global parameters	defines internal database	defines facts and rules	defines the body of the program	drives execution

Examples shown in the lecture

Prolog Structure

A Prolog program structure consists of five segments, not all of them must appear in each program. The following segments must be included in each program: **predicates, clauses, and goal**. The five segments of the Prolog structure are:

1. Domains

To define global parameters used in the program.

Example:

```
domains
i = integer
c = char
s = string
r = real
```

2. Database

To define the internal database generated by the program. Example:

```
database
greater(integer)
```

.

3. Predicates:

to define facts and rules used in the program. predicates mark(symbol, integer).

4. Clauses:

to define the body of the program. For the above predicates, the clauses portion may contain: clauses mark(a, 20).

5. Goal:

can be internal or external, the internal goal is written after clauses portion, but the external goal is supported by the prolog compiler if the program syntax is correct, this portion contains the rule that drives the program execution

6. Basic Elements and the Knowledge Base

The lecture states that there are only three basic constructs in Prolog: facts, rules, and questions. A collection of facts and rules is called a knowledge base (or a database).

Basic Elements of Prolog

There are only three basic constructs in Prolog: **Facts, Rules, and Questions**.
A collection of facts and rules is called a **knowledge base (or a database)**.

That is, Prolog programs are simply **knowledge bases**, collections of facts and rules which describe some collection of relationships that we find interesting.

Interpretation: Prolog programs are knowledge bases that describe collections of relationships that we find interesting.

7. Key Takeaways and Sources

• Prolog is introduced here as a declarative, logic-based language.

• Its early foundations include terms, variables, data types, structure, and knowledge bases.

• The first nine pages already establish why Prolog is useful in AI-oriented courses.

Primary content source: uploaded lecture PDF, pages 1–9.

Visual asset links:

Unsplash — AI chip image

Unsplash — code screen image

Unsplash — programmer desk image

Wikimedia Commons — flowchart image

Slideshare — knowledge graph image