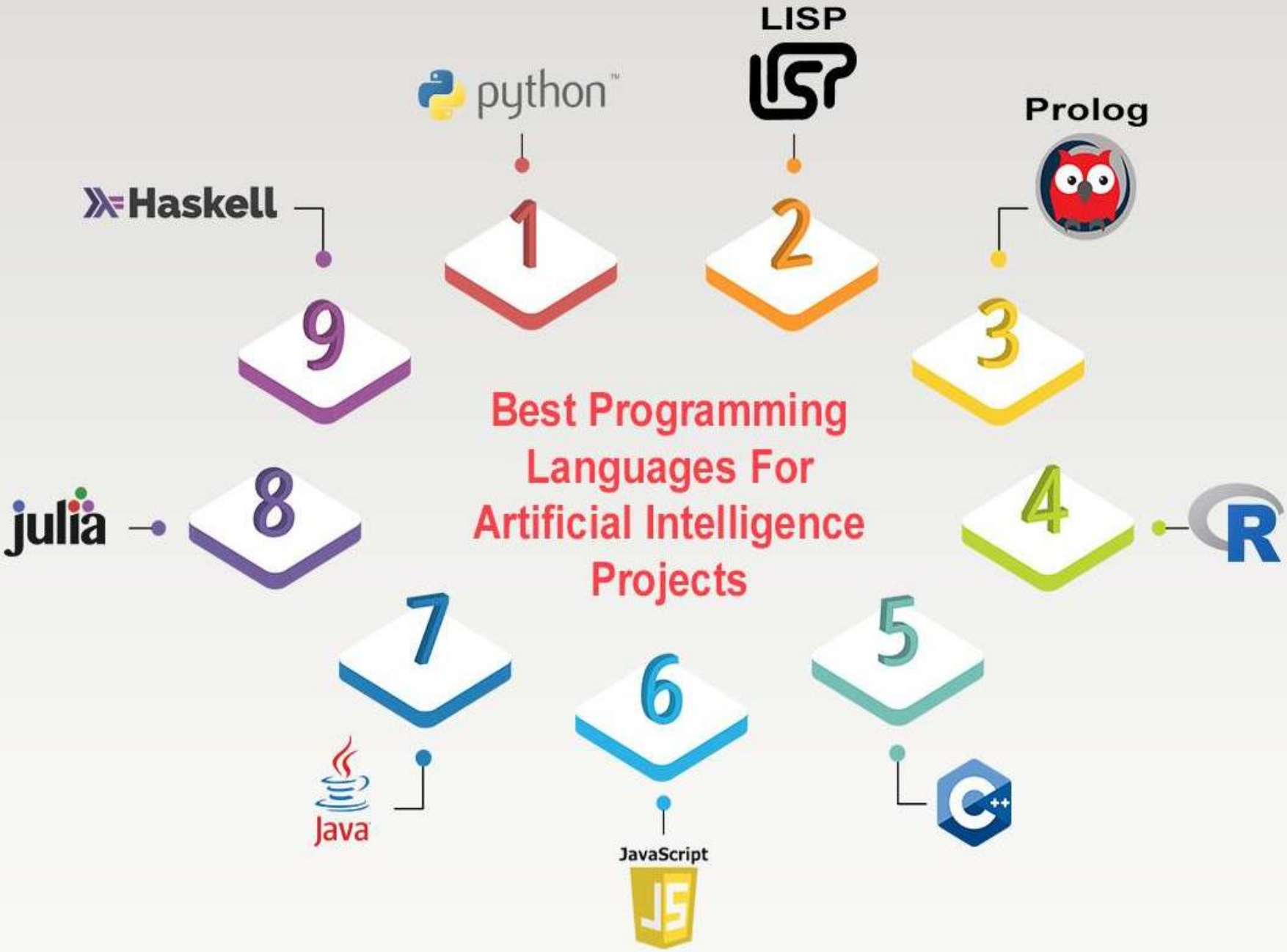

LISP
python
Prolog
Haskell
1
2
3
9
Best Programming Languages For Artificial Intelligence Projects
julia
8
4
R
7
6
5
Java
JavaScript
C++

Prolog programming
Introduction to Prolog

CSE 4102

Presented By:
Rahat Khan
Lecturer, Dept. of CSE
Khwaja Yunus Ali University

Prolog

"Programming with Logic"

Very different from other programming languages

- Declarative (not procedural)
- Non Recursive

(no “for” or “while” loops)

- Relations (no functions)

Differences between Procedural P. and Logic P.

- *Architecture*:Von Neumann machine (sequential steps)
- *Syntax:* Sequence of statements (a, s, I)
- *Computation:* Sequential statements execution
- *Control*: Logic and control are mixed together
- Abstract model (dealing with objects and their relationships)
- Logic formulas (Horn Clauses)
- Deduction of the clauses
- Logic and control can be separated

Prolog(high level language) logically deduces new facts about the situation we described.

SWI-Prolog (IDE + Interpreter)

Similar IDE: Visual Prolog/GNU Prolog/Inter Prolog
- developed by Jan Wielemaker.
- Platform Independent

SWI (Software Improvement)- Prolog editor is an open source implementation of the programming language . Prolog is a free software and you can download it from here:

http://www.swi-prolog.org/Download.html

Getting started:

Any Prolog program in SWI-Prolog is written in a file with extension .pl.

Now, time for environment setup..

Facts

The syntax of facts is the same as in the predicate calculus.

Examples: Fact_name(parameter).

```
fish(ray).              % Ray is a fish.
fish(salmon).           % Salmon is a fish.
red(car).               % Car is red.
likes(mary , john).     % Mary likes John.
factorial(3 , 6).       % the factorial of 3 is 6
```

Add this fact in knowledge base.

*Fact must writes in lower case.

Prolog is conversational

- Here is an illustrative conversation with a Prolog system:

  ?- fish(ray).

  yes

  ?- fish(dog).

  no

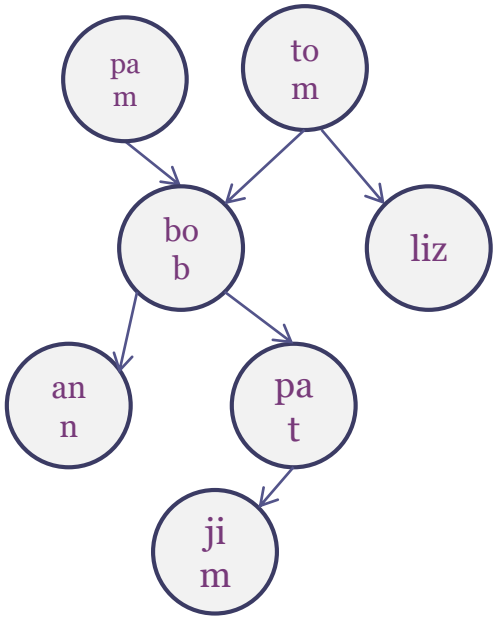
  ?- fish(sardine).

  no

- A sardine is a fish, but prolog doesn't know that unless it is specified as a fact in the source file.

Defining relations by facts

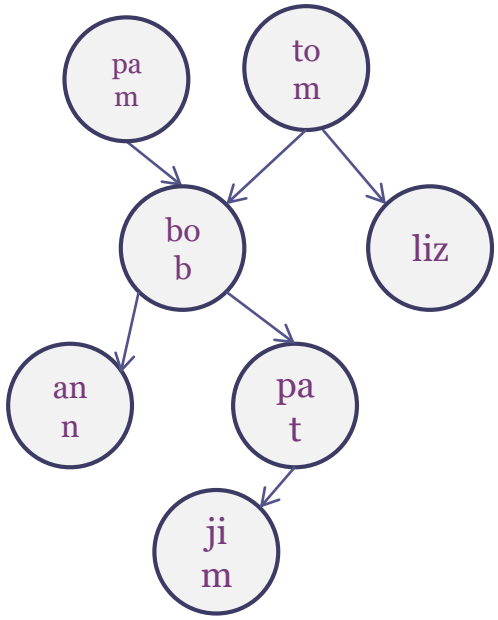
- The whole family tree can be described by the following facts:



Defining relations by facts

- The whole family tree can be described by the following facts:

```
parent(pam,bob).
parent(tom,bob).
parent(tom,liz).
parent(bob,ann).
parent(bob,pat).
parent(pat,jim).
```



Defining relations by facts

We can also add information about the origin of the people by the following facts:

female(pam).
male(tom).
male(bob).
female(liz).
female(pat).
female(ann).
male(jim).

** Here
female(pam) is a **Unary relation**
parent(bob,pat) is a **Binary relation**.

How to write Rule:

Inference

- Given the statement:

  "Fido is a dog"

  "All dogs are animals." and

  "All animals will die."

- Prove that "Fido will die." ?

Inference

"Fido is a dog"

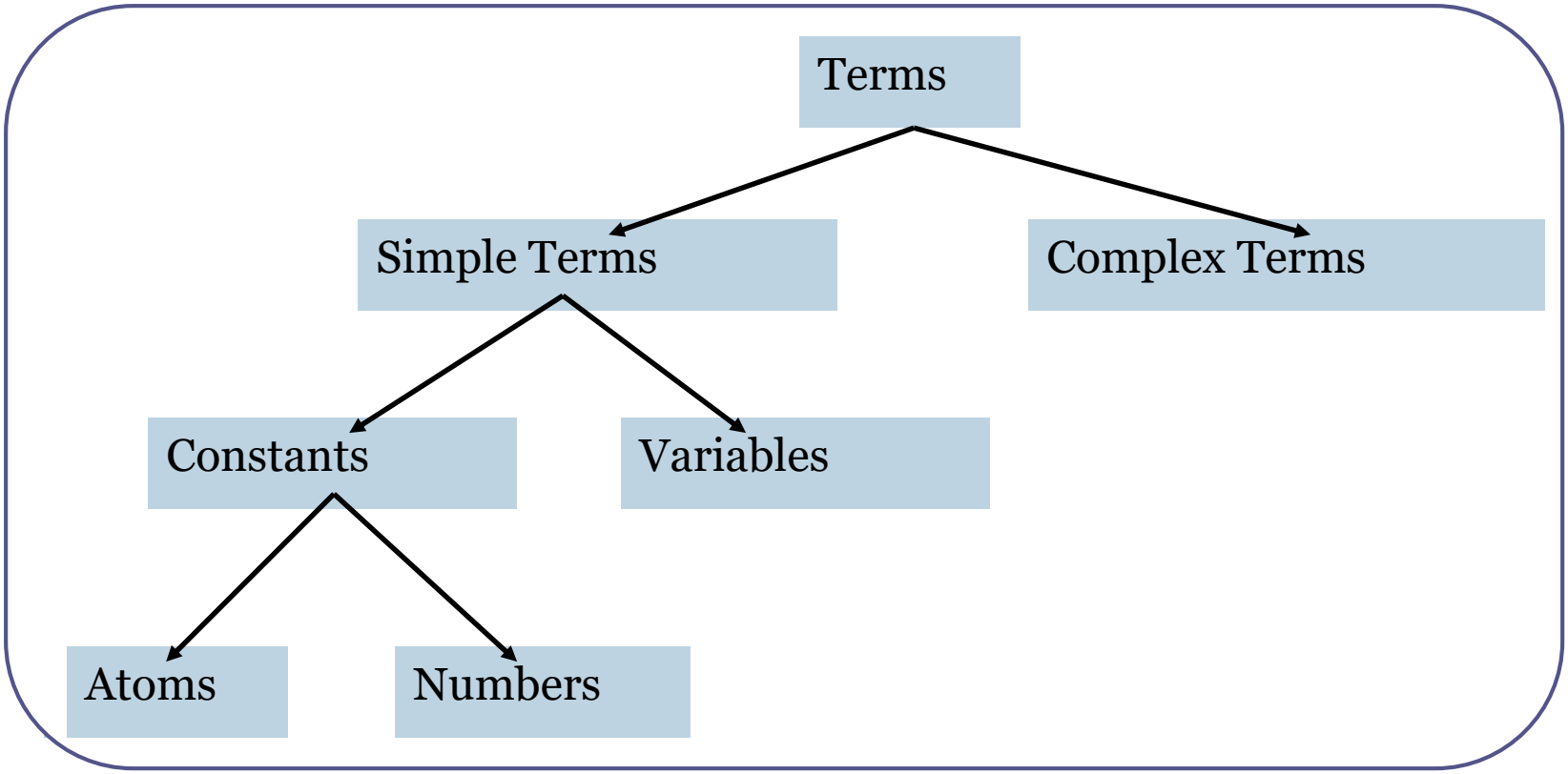
"All dogs are animals." and

"All animals will die."

```
dog(fido).
animal(X):-dog(X).
die(X):-animal(X).
```

Prolog Syntax

What exactly are facts, rules and queries built out of?



Variables

- A sequence of characters of upper-case letters, lower-case letters, digits, or underscore, starting with either an uppercase letter or an underscore. Examples:

  X, Y, Variable, Vincent, _tag

Atoms

A sequence of characters of upper-case letters, lower-case letters, digits, or underscore, starting with a lowercase letter

- *Examples*: **butch**, **big_kahuna_burger**, **playGuitar**

An arbitrary sequence of characters enclosed in single quotes

- *Examples*: **'Vincent'**, **'Five dollar shake'**, **'@$%'**

A sequence of special characters

- *Examples*: **: , ; . :-**

Numbers

- Integers:

  12, -34, 22342

- Floats:

  34573.3234, 0.3435

Complex Terms

- Atoms, numbers and variables are building blocks for **complex terms**
- Complex terms are built out of a **functor** directly followed by a sequence of **arguments**
  - Arguments are put in round brackets, separated by commas
  - The functor must be an atom

Examples of complex terms

- Examples we have seen before:
  - playsAirGuitar(mira)
  - loves(mira, ratul)
  - jealous(ratul, himu)

- Complex terms inside complex terms:
  - hide(X,father(father(father(butch))))

Arity

- The number of arguments a complex term has is called its arity

- Examples:

woman(himu) is a term with arity 1
loves(mira,ratul) has arity 2
father(father(ratul)) arity 1

Arity is important

- In Prolog you can define two predicates with the same functor but with different arity
- Prolog would treat this as two different predicates
- In Prolog documentation arity of a predicate is usually indicated with the suffix "/" followed by a number to indicate the arity

Exercises

- Exercise 1.1 Which of the following sequences of characters are atoms, which are variables, and which are neither?
    1. vINCENT
    2. Footmassage
    3. Variable23
    4. Variable2000
    5. big_kahuna_burger
    6. ’big kahuna burger’
    7. big kahuna burger
    8. ‘Jules’
    9. _Jules
    10. ’_Jules’

Exercise

- Which of the following queries are satisfied? Where relevant, give all the variable instantiations that lead to success.
    1. ?- magic(house_elf).
    2. ?- wizard(harry).
    3. ?- magic(wizard).
    4. ?- magic('McGonagall').
    5. ?- magic(Hermione).