

CS339: Abstractions and Paradigms for Programming

Logic Paradigm

Manas Thakur
CSE, IIT Bombay



Autumn 2024

Declarative vs Operational

➤ Square root of x:

➤ A number y such that $y^2 = x$ Declarative

➤ Newton's method

Operational

```
(define (sqrt x)
  (define (good-enough? guess)
    (< (abs (- (square guess) x)) 0.001))
  (define (improve guess)
    (average guess (/ x guess)))
  (define (sqrt-iter guess)
    (if (good-enough? guess)
        guess
        (sqrt-iter (improve guess))))
  (sqrt-iter 1.0))
```



Sum a list of numbers: Imperative (Java)

```
int sum(int[] list) {  
    int result = 0;  
    for (int i = 0; i < list.length; i++) {  
        result += list[i];  
    }  
    return result;  
}
```



Sum a list of numbers: Functional (Scheme)

```
(define (sum list)
  (if (null? list)
      0
      (+ (car list) (sum (cdr list)))))
```

Sum a list of numbers: Logic (Prolog)

```
sum([], 0).  
sum([H | T], N) :- sum(T, M), N is H + M.
```

Functional is sometimes close to declarative, but logic is closer.

Logic Paradigm

- Programs consist of just facts and rules.
- Not necessary to describe the “procedure” or the control flow at a very low-level.
- Who does the computation then?
- In other words, who has the onus of translating the “declarative” description to an “algorithm” that computes on the von-Neumann architecture?
 - The Interpreter!



Logic Paradigm: Usage

- **Prolog** quite popular in rule-based Artificial Intelligence.
- **Datalog** becoming very popular in program analysis, code optimization, and type inference.
- **SQL** already the de-facto of relational databases.



Predicates and Horn Clauses

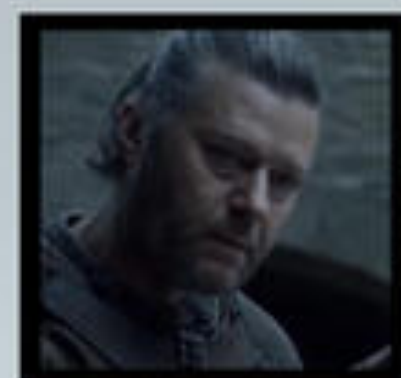
- If it is precipitating in a city C and the temperature in C is freezing, then it is snowing in C.
 - `snowing(C); precipitation(C); freezing(C)` **Predicates**
 - `snowing(C) <- precipitation(C) AND freezing(C)` **Horn clauses**
- When does it snow at Mumbai?
 - Instantiate the variables.







House: STARK



RICKARD
STARK



UNKNOWN



BRANDON
STARK



EDDARD
STARK



CATELYN
TULLY



RHAEGAR
TARGARYEN



LYANNA
STARK



BENJEN
STARK



ROBB
STARK



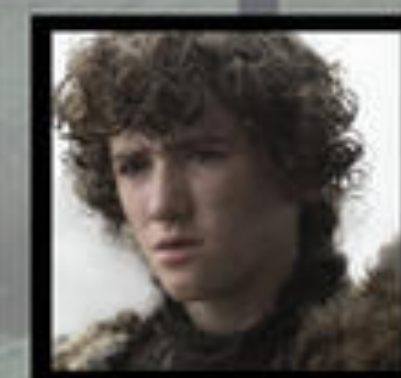
SANSA
STARK



ARYA
STARK



BRAN
STARK



RICKON
STARK



JON
SNOW

Toxic
Chemistry ©

(The Shortest?) Introduction to Prolog

➤ Two kinds of *terms*:

➤ Facts

- `father(ned, arya).`
- `mother(catelyn, bran).`

➤ Rules

- `parent(X, Y) :- father(X, Y).`
- `parent(X, Y) :- mother(X, Y).`
- `grandparent(X, Z) :- parent(X, Y), parent(Y, Z).`

Rules of the game (*aka Syntax*):

- Constants start with small letters.
- Variables start with capital letters.
- Full stop necessary after each fact/rule.
- No space before the opening parenthesis.
- Multiple terms with the same *head* indicate disjunction.
- A comma between terms indicates conjunction.

Querying in Prolog

- ?- father(ned, sansa).
- ?- grandparent(rickard, bran).

Homework: Define rules
cousin, uncle, aunt, sibling.

Closed World Assumption

- Inferences can be drawn only from known facts.

