Faculty of Computing

Name: Bushra Nawaz

SapId: 40283

Artificial Intelligence - fall 2024

(LAB -10)

Instructor

Rehana Sanam,

TF, Faculty of Computing,

Riphah International University, Islamabad

Lab 10 Tasks

Question 01:

1. Which of the following sequences of characters are atoms, which are variables, which are complex terms, and which are not terms at all? Give the functor and arity of each complex term.
    a. loves(Vincent,mia)
    b. 'loves(Vincent,mia)'
    c. Butch(boxer)
    d. boxer(Butch)
    e. and(big(burger), kahuna(burger))
    f. and(big(X), kahuna(X))
    g. _and(big(X), kahuna(X))
    h. (Butch kills Vincent)

a. loves(Vincent, mia)

- **Type:** Complex Term
- **Functor:** loves
- **Arity:** 2 (it takes two arguments)
- **Explanation:** This is a valid complex term, where `loves` is the functor, and `Vincent` and `mia` are arguments. `Vincent` and `mia` are atoms in this case.

---

b. 'loves(Vincent,mia)'

- **Type:** Atom
- **Explanation:** This is an atom because it is a string of characters enclosed in single quotes. The string itself represents a constant value, not a term with a functor and arguments. This is not a complex term.

---

c. Butch(boxer)

- **Type:** Complex Term
- **Functor:** Butch
- **Arity:** 1 (it takes one argument)
- **Explanation:** `Butch` is the functor, and `boxer` is the argument. `boxer` is an atom in this case.

---

d. boxer(Butch)

- **Type:** Complex Term
- **Functor:** boxer
- **Arity:** 1 (it takes one argument)
- **Explanation:** `boxer` is the functor, and `Butch` is the argument. `Butch` is an atom in this case.

---

e. and(big(burger), kahuna(burger))

- **Type:** Complex Term
- **Functor:** and
- **Arity:** 2 (it takes two arguments)
- **Explanation:** This is a complex term where `and` is the functor and it has two arguments. The arguments themselves are complex terms: `big(burger)` and `kahuna(burger)`, where both `big` and `kahuna` are functors, and `burger` is an atom.

---

f. and(big(X), kahuna(X))

- **Type:** Complex Term
- **Functor:** and
- **Arity:** 2 (it takes two arguments)
- **Explanation:** This is a complex term where `and` is the functor. The arguments `big(X)` and `kahuna(X)` are also complex terms, where `X` is a variable. The functors are `big` and `kahuna`, and the variable `X` is the argument.

---

g. _and(big(X), kahuna(X))

- **Type:** Not a Term
- **Explanation:** This is **not a valid term** in many logic programming languages like Prolog. The underscore (_) at the beginning of `and` makes it an invalid identifier. In Prolog, a valid term cannot start with an underscore unless it is used in a special way (e.g., as an anonymous variable).

---

h. (Butch kills Vincent)

- **Type:** Not a Term
- **Explanation:** This is a phrase or a sentence in natural language, not a valid logical term. In a formal language like Prolog, terms need to follow a specific structure (functor with arguments), and this is not one.

---

i. kills(Butch Vincent)

- **Type:** Not a Term
- **Explanation:** This is **not a valid term** because it lacks a comma separating the arguments. A valid term would be something like `kills(Butch, Vincent)`.

j. kills(Butch, Vincent)

- **Type:** Complex Term
- **Functor:** kills
- **Arity:** 2 (it takes two arguments)
- **Explanation:** This is a valid complex term. `kills` is the functor, and `Butch` and `Vincent` are the arguments, both of which are atoms.

2. How many facts, rules, clauses, and predicates are there in the following knowledge base?
What are the heads of the rules, and what are the goals they contain?
woman(vincent). woman(mia). man(jules). person(X):- man(X); woman(X).
loves(X,Y):- father(X,Y). father(Y,Z):- man(Y), son(Z,Y).
father(Y,Z):- man(Y), daughter(Z,Y).

Solution:

Facts:

woman(vincent).
woman(mia).
man(jules).
There are 3 facts in total.
Rules:
person(X) :- man(X); woman(X).
loves(X, Y) :- father(X, Y).
father(Y, Z) :- man(Y), son(Z, Y).
father(Y, Z) :- man(Y), daughter(Z, Y).
There are 4 rules in total.
Clauses:
Each fact is a single clause.
Each rule also represents one clause.
Total number of clauses = 3 (facts) + 4 (rules) = 7 clauses.
Predicates: The unique predicates in the knowledge base are:
woman/1
man/1
person/1
loves/2
father/2
son/2
daughter/2
This gives us 7 predicates.
Heads of the Rules and Goals
Let's identify the heads of each rule and the goals within each rule.
Rule 1: person(X) :- man(X); woman(X).
Head: person(X)
Goals: man(X), woman(X)

Rule 2: loves(X, Y) :- father(X, Y).
Head: loves(X, Y)
Goal: father(X, Y)
Rule 3: father(Y, Z) :- man(Y), son(Z, Y).
Head: father(Y, Z)
Goals: man(Y), son(Z, Y)
Rule 4: father(Y, Z) :- man(Y), daughter(Z, Y).
Head: father(Y, Z)
Goals: man(Y), daughter(Z, Y)

3. Represent the following in Prolog:
    a. Butch is a killer.
    b. Mia and Marsellus are married.
    c. Zed is dead.
    d. Marsellus kills everyone who gives Mia a foot massage.
    e. Mia loves everyone who is a good dancer.
    f. Jules eats anything that is nutritious or tasty.

    Solution:

    a. Butch is a killer.

```
killer(butch).
```

    b. Mia and Marsellus are married.

```
married(mia, marsellus).
married(marsellus, mia).  % Assuming marriage is bidirectional
```

    c. Zed is dead.

```
dead(zed).
```

    d. Marsellus kills everyone who gives Mia a foot massage.

```
kills(marsellus, X) :- gives_foot_massage(X, mia).
```

    e. Mia loves everyone who is a good dancer.

```
loves(mia, X) :- good_dancer(X).
```

    f. Jules eats anything that is nutritious or tasty.

```
eats(jules, X) :- nutritious(X).
eats(jules, X) :- tasty(X).
```

4. Suppose we are working with the following knowledge base:

```
wizard(ron). hasWand(harry). quidditchPlayer(harry). wizard(X):-
hasBroom(X), hasWand(X).
hasBroom(X):- quidditchPlayer(X).
```

How does Prolog respond to the following queries?

a. wizard(ron).
b. witch(ron).
c. wizard(hermione).
d. witch(hermione).
e. wizard(harry).
f. wizard(Y).
g. witch(Y).

Solution:

Query Prolog's Response

a. wizard(ron). Yes

b. witch(ron). No

c. wizard(hermione). No

d. witch(hermione). No

e. wizard(harry). Yes

f. wizard(Y). `Y = harry`

g. witch(Y). No

Evaluation criteria

The evaluation criteria for this lab will be based on the completion of the following tasks. Each task is assigned the marks percentage which will be evaluated by the instructor in the lab whether the student has finished the complete/partial task(s).

Evaluation of the Lab 10

Task No	Description	Marks
1	Task 1	10

Further Reading

The slides and reading material can be accessed from the folder of the class instructor available at moellim.