
Table of Contents

.....	1
Part 1: Propositional Logic	1
Part 2: Semantic Network (Objects and Relationships)	2
Part 3: Simple Logical Reasoning	3

```
% MATLAB Code for Knowledge Representation in AI
clc;
clear;
close all;
```

Part 1: Propositional Logic

We will represent some simple facts and use logical operators for reasoning.

```
% Define propositions
% p: "It is raining."
% q: "The ground is wet."
% r: "There is a rainbow."

% Initial Facts
raining = true;    % It is raining
wet_ground = false; % Initially, the ground is not wet
rainbow = false;    % Initially, no rainbow

% Represent the logical relationships:
% If it rains, the ground becomes wet.
if raining
    wet_ground = true; % If it's raining, the ground is wet
end

% If the ground is wet, and it's raining, there is a rainbow.
if wet_ground && raining
    rainbow = true; % If both conditions hold, there is a rainbow
end

% Display the facts
fprintf('Is it raining? %s\n', mat2str(raining));
fprintf('Is the ground wet? %s\n', mat2str(wet_ground));
fprintf('Is there a rainbow? %s\n', mat2str(rainbow));

Is it raining? true
Is the ground wet? true
Is there a rainbow? true
```

Part 2: Semantic Network (Objects and Relationships)

```
% Let's create a simple semantic network using a structure array to represent
objects and their properties.

% Define some objects in the environment
objects = struct();

% Define objects
objects.house = struct('type', 'building', 'hasWindows', true, 'hasRoof',
true);
objects.tree = struct('type', 'plant', 'hasLeaves', true, 'height', 20); % %
height in meters
objects.car = struct('type', 'vehicle', 'hasWheels', true, 'color', 'red');
objects.person = struct('type', 'human', 'hasEyes', true, 'height', 1.75); %

height in meters

% Query about relationships (e.g., Does the person have eyes?)
fprintf('\nDoes the person have eyes? %s\n', mat2str(objects.person.hasEyes));

% Query about the tree's height
fprintf('The height of the tree is %.2f meters.\n', objects.tree.height);

% Query about the car's color
fprintf('The color of the car is %s.\n', objects.car.color);

% Display object properties in the command window
fprintf('\nObject Properties:\n');
fields = fieldnames(objects);
for i = 1:length(fields)
    fprintf('\nObject: %s\n', fields{i});
    obj = objects.(fields{i});
    disp(obj);
end
```

*Does the person have eyes? true
The height of the tree is 20.00 meters.
The color of the car is red.*

Object Properties:

*Object: house
 type: 'building'
 hasWindows: 1
 hasRoof: 1*

*Object: tree
 type: 'plant'
 hasLeaves: 1*

```
height: 20

Object: car
    type: 'vehicle'
hasWheels: 1
color: 'red'

Object: person
    type: 'human'
hasEyes: 1
height: 1.7500
```

Part 3: Simple Logical Reasoning

We will add some simple reasoning based on relationships in our semantic network.

```
% For example:
% If the person has eyes, we can reason that the person can see.
if objects.person.hasEyes
    canSee = true; % The person can see because they have eyes
else
    canSee = false; % The person cannot see
end

fprintf('\nCan the person see? %s\n', mat2str(canSee));

% Another reasoning example:
% If the car has wheels, we can infer that the car can move.
if objects.car.hasWheels
    canMove = true; % The car can move because it has wheels
else
    canMove = false; % The car cannot move
end

fprintf('Can the car move? %s\n', mat2str(canMove));
```

Can the person see? true
Can the car move? true

Published with MATLAB® R2024b