**Experiment – 3**

**Monkey and Banana Problem**

The monkey and banana problem are often used as a simple example of problem solving.

We will use the following variations of the problem

* There is a monkey at the door inside a room
* In the middle of the room a banana is hanging from the ceiling
* The monkey is hungry and want to get the banana, but he cannot stretch his enough from the floor
* At the window of the room there is a box the monkey may use

Diagram

Description automatically generated

The monkey can perform the following actions

1. Walk on the floor
2. Push the box
3. Climb the box
4. Grasp the banana if it is standing on the box directly under the banana.

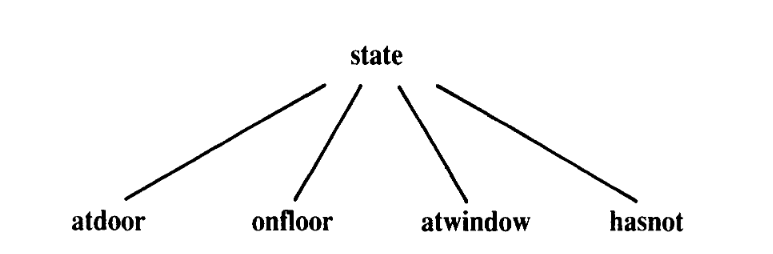
Can the monkey get the banana?

* One important task in the programming is that of finding a representation of the problem in terms of concepts of the programming language used.
* In our case we can think of the monkey world as always being in some state that can change in time
* The Current State is determined by the positions of the object

For Example: The initial state of the world is determined by

1. Monkey is at door
2. Monkey is on floor
3. Box is at window
4. Monkey does not have a banana

State(\_\_\_\_\_\_,\_\_\_\_\_\_,\_\_\_\_\_\_,has)



There are four types of moves

1. Walk Around
2. Push Box
3. Climb Box
4. Grasp Banana

**Prolog Program**

on(floor,monkey).

on(floor,chair).

in(room,monkey).

in(room,chair).

in(room,banana).

at(ceiling,banana).

strong(monkey).

grasp(monkey).

climb(monkey,chair).

push(monkey,chair):-

strong(monkey).

under(banana,chair):-

push(monkey,chair).

canreach(banana,monkey):-

at(floor,banana);

at(ceiling,banana),

under(banana,chair),

climb(monkey,chair).

canget(banana,monkey):-

canreach(banana,monkey),grasp(monkey).

Output Graphical user interface, text, application

Description automatically generated