

TD2: Prolog

Exercise 1

Assume given a set of facts of the form

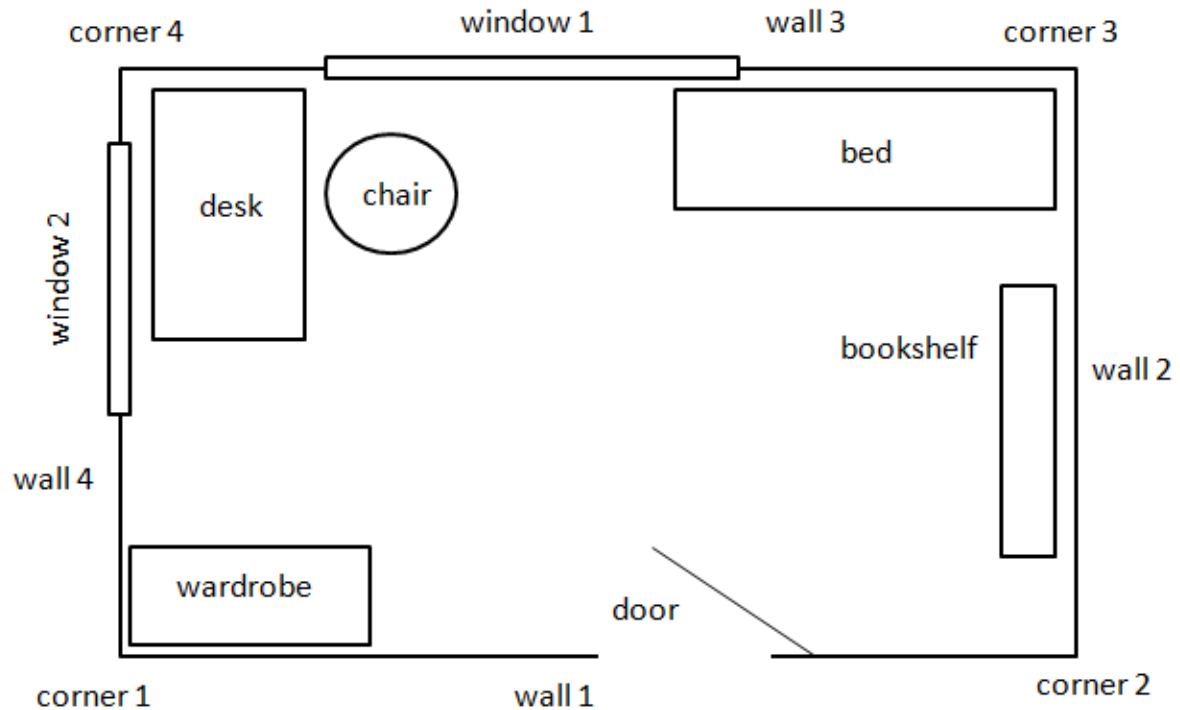
`father(name1, name2)` (`name1` is the father of `name2`).

1. Define a predicate `brother(X, Y)` which holds iff `X` and `Y` are brothers.
2. Define a predicate `cousin(X, Y)` which holds iff `X` and `Y` are cousins.
3. Define a predicate `grandson(X, Y)` which holds iff `X` is a grandson of `Y`.
4. Define a predicate `descendent(X, Y)` which holds iff `X` is a descendent of `Y`.

Solution:

```
brother(X,Y) :- father(Z,X), father(Z,Y).  
cousin(X,Y) :- father(Z,X), father(W,Y),  
brother(Z,W).  
grandson(X,Y) :- father(Z,X), father(Y,Z).  
descendent(X,Y) :- father(Y,X).  
descendent(X,Y) :- father(Z,X), descendent(Z,Y).
```

Exercise 2



Represent this student's room such that the following queries are possible:

1. Which furniture is in the room?
2. How many doors, windows, tables, ... are in the room?
3. Where is the table, the chair, ... ?
4. What is to the left (right) of the table, ...?
5. What is at the wall 2, at the window 1, ... ?
6. What is in the corner 1, ... ?

Solution

furniture(wardrobe, 1).

furniture(bookshelf, 1).

furniture(bed, 1).

furniture(desk, 1).

furniture(chair, 1).

furniture(X) :- furniture(X, N).

/* Query 1: ?- furniture(X) */

number(X, N) :- furniture(X, N).

/* Query 2: ?- number(Object, Number) */

number(door, 1).

number(window, 2).

at(wall_1, wardrobe).

/* Query 5: ?- at(At, Object) */

at(wall_1, door).

at(wall_2, bookshelf).

at(wall_2, bed).

at(wall_3, bed).

at(wall_3, window_1).

at(wall_3, desk).

at(wall_4, desk).

at(wall_4, window_2).

at(wall_4, wardrobe).

at(window_1, bed).

at(window_2, desk).

in(corner_1, wardrobe).

/* Query 6: ?- in(Place, Object) */

in(corner_3, bed).

in(corner_4, desk).

place(X, P) :- at(P, X).

/* Query 3: ?- place(Object, Place) */

place(X, P) :- in(P, X).

place(chair, at_desk).

left(door, bookshelf).

/* Query 4: ?- left(Object1, Object2) */

left(bookshelf, bed).

left(bed, desk).

left(desk, wardrobe).

left(wardrobe, door).

right(X, Y) :- left(Y, X).

/* Query 4: ?- right(Object1, Object2) */