Master MLDM/DSC/CPS2 - 2020/2021 - First year
Introduction to Artificial Intelligence - Exam on Prolog

Maximum time allocated: 3h00 - No documents authorized. Your copy must be written in English. Grading will depend on the cleanliness of your copy and the clarity of your explanations. TAKE CARE: any cheating will be severely punished and will lead to a formal complaint to the disciplinary council of the university.

1 Proof tree (≃ 4 points)

Consider the Prolog program below:

```
p1(A,B) :- r(X), q(A), t(X,B).
p1(A,B) :- s(A), t(B,A).
p1(A,B) :- p2(A,X), t(X,B).

p2(X,Y) :- q(Y), r(X).
p2(X,Y) :- s(X), q(Y).

s(b).     s(a).     s(e).    s(d).
q(42).    q(21).
r(a).     r(b).     r(e).
t(42,b).  t(21,e). t(21,d).
```

1. Draw the proof tree of the resolution of the goal: `?- p1(e,X).` and give all the solutions for this goal.
2. Suppose we put a cut between `q(A)` and `t(X,B)` in the first clause of the program and another cut between `q(Y)` and `r(X)` in the fourth clause of the program. Show, on the tree you built at the previous question, which branches are pruned during the resolution of the goal: `?- p1(e,X).` and give again all the solutions for this goal.

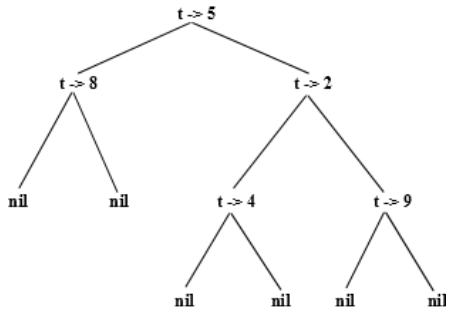
2 Unification (≃ 2 points)

Suppose the occurs check is activated. For each of the Prolog goals below, say whether it is true or false. In cases where it makes sense, give the value(s) of the variable(s) that make(s) a goal true.

1. `?- [a, X, b, c] = [a, b, c].`
2. `?- p(f(a), f(Y)) = p(f(X), f(1+2)).`
3. `?- p(a, g(X), Y) = p(Y, g(f(X)), a).`
4. `?- [a,b,c,d,e,f] = [a, X, c, Y | Z].`
5. `?- X=2, X=3.`
6. `?- x(a,f(X,g(a,Y),Z,W),b,T) = x(X,f(a,T,c,b),Y,g(X,W)).`
7. `?- X = 2 + 1.`
8. `?- sentence([det(a),adj(little),adj(big),noun(dog)]) = sentence([X,adj(Y)|Z]).`

3 Trees (≃ 3 points)

A tree can be represented in Prolog by a compound term t(V,L,R) where V is the value of the root of the tree, L is the left subtree of the tree, and R is the right subtree of the tree. The empty tree is denoted `nil`.
For example the compound term t(5,t(8,nil,nil),t(2,t(4,nil,nil),t(9,nil,nil)) is a Prolog representation of the following tree:



Define the following Prolog predicates :

1. `is_a_tree/1` where `is_a_tree(T)` is true if T is an empty tree denoted `nil` or if T is a tree of the form t(V,L,R) where L and R are trees.

2. `count_leaves/2` where `count_leaves(T,N)` is true if N is the number of leaves of the tree T.
3. `collect_leaves/2` where `collect_leaves(T,L)` is true if L is the list containing all the leaves of the tree T.

(in the tree above, there are three leaves: 8, 4 and 9)

4 Lazy evaluation (≂ 2 points)

The Jack function is defined by:

$$jack(m,p) = \begin{cases} 1 & if\ m = 1\ or\ p = 1 \\ m & if\ p = 0 \\ p & if\ m = 0 \\ jack(m-1,p-1) * jack(m-2,p-2) & if\ m > 1\ and\ p > 1 \end{cases}$$

1. Write the basic Prolog program that defines a predicate `jack/3` where `jack(M,P,Res)` is true if `Res` is the result of the Jack function of `M` and `P`.
2. Write the Prolog program that defines the predicate `lazyJack/3` where `lazyJack(M,P,Res)` is true if `Res` is the result of the Jack function of `M` and `P` but the Jack function is calculated in a lazy way, i.e. once we have calculated a value for `lazyJack(M,P,Res)` we want to store this information so that it can be reused in the calculation of another value for `lazyJack(M',P',Res')`.

5 Knowledge base modeling and querying (≂ 3 points)

Convert the following information into a Prolog program.

- If a person `P1` is richer than another person `P2`, and they live in the same place then `P1` pays more taxes than `P2`.
- If a person `P1` is richer than another person `P2`, and `P1` lives in town and `P2` lives in the countryside, then `P1` is smarter than `P2`.
- If a person `P1` is richer than a person `P3` that is richer than a person `P2`, then `P1` is richer than `P2` (the relationship "richer than" is transitive).
- The relationship "younger than" is also transitive.
- If a person `P1` is smarter than a person `P2`, or if `P1` is younger than `P2` and `P1` is a student and `P2` is a teacher, then `P1` is happier than `P2`.
- If `P1` pays more taxes than `P2` then `P1` is jealous of `P2`.
- If `P1` is happier than `P2` or `P1` is jealous of `P2`, then `P1` has feelings.
- Bess and Dana live in the same place.
- John is a student.
- John is younger than Mary.
- Suzy is a teacher.
- Cody lives in town.
- Cody is richer than Dana.
- Dana lives in the countryside.
- Bess is richer than Cody.
- Mary is younger than Suzy.

Considering the Prolog program you just wrote, explain how you can find all the persons that have feelings.

6 DCG (≂ 6 points)

Write a DCG in Prolog that can be used to convert a number written with digits (at most four in this exercise) to the number written with letters and convert a number written with letters to a number written with digits (at most four in this exercise). For example, you should be able to run the following goals:

```
?- phrase(number(2328),S).
S = [two,thousand,three,hundred,and,twenty,eight]

?- phrase(number(N),[two,thousand,three,hundred,and,twenty,eight]).
N = 2328
```

More generally, give the Prolog clause generated from the following DCG rule after loading it into the Prolog workspace: `p(X) --> s(X,Y), [a], {N is Y+1}, t(N), [b].`