Santhoshkrishnachaitanya Chelikavada

N1604733G

CZ 3005 – Lab 2 – TS3

Exercise 1 – The Smart Phone Rivalry

1)

company(sumsum)

company(appy)

smartphonetech(galacticas3)

develop(galacticas3,sumsum)

boss(stevey)

competitor(sumsum,appy)

stole(stevey,galacticas3,sumsum)

∀X competitor(X,appy) => rival(X)

∀X smartphonetech(X) => business(X)

∀X,Y,Z unethical(X) => boss(X) ∧ business(Y) ∧ rival(Z) ∧  stole(X,Y,Z)

2)

company(sumsum).

company(appy).

competitor(sumsum,appy).

develop(galacticas3,sumsum).

smartphonetech(galacticas3).

boss(stevey).

stole(stevey,galacticas3,sumsum).

rival(X) :- competitor(X,appy).

business(X) :- smartphonetech(X).

unethical(X):- boss(X), business(Y), company(Z),rival(Z), stole(X,Y,Z).

3)

[trace] 3 ?- unethical(stevey).

Call: (7) unethical(stevey) ? creep

Call: (8) boss(stevey) ? creep

Exit: (8) boss(stevey) ? creep

Call: (8) business(\_G517) ? creep

Call: (9) smartphonetech(\_G517) ? creep

Exit: (9) smartphonetech(galacticas3) ? creep

Exit: (8) business(galacticas3) ? creep

Call: (8) company(\_G517) ? creep

Exit: (8) company(sumsum) ? creep

Call: (8) rival(sumsum) ? creep

Call: (9) competitor(sumsum, appy) ? creep

Exit: (9) competitor(sumsum, appy) ? creep

Exit: (8) rival(sumsum) ? creep

Call: (8) stole(stevey, galacticas3, sumsum) ? creep

Exit: (8) stole(stevey, galacticas3, sumsum) ? creep

Exit: (7) unethical(stevey) ? creep

true

Exercise 2:



Knowledge Base:

male(charles).

male(andrew).

male(edward).

female(ann).

female(elizabeth).

monarch(elizabeth).

queen(elizabeth).

offspring(charles,elizabeth).

offspring(ann,elizabeth).

offspring(andrew,elizabeth).

offspring(edward,elizabeth).

elder\_than(charles,ann).

elder\_than(ann,andrew).

elder\_than(andrew,edward).

%The succession rules are managed through OR branches.

succession(X) :- offspring(X,Z),monarch(Z),male(X), elder\_than(X,Y).

succession(X) :- offspring(X,Z),monarch(Z),male(X), not(elder\_than(X,Y)).

succession(X) :- offspring(X,Z),monarch(Z),female(X), elder\_than(X,Y).

succession(X) :- offspring(X,Z),monarch(Z),female(X), not(elder\_than(X,Y)).

[debug] ?- succession(X).

X = charles ;

X = andrew ;

X = edward ;

X = ann ;

false.

Trace -

[trace] ?- succession(X).

Call: (8) succession(\_582) ? creep

Call: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(charles, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) male(charles) ? creep

Exit: (9) male(charles) ? creep

Call: (9) elder\_than(charles, \_798) ? creep

Exit: (9) elder\_than(charles, ann) ? creep

Exit: (8) succession(charles) ? creep

X = charles ;

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(ann, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) male(ann) ? creep

Fail: (9) male(ann) ? creep

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(andrew, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) male(andrew) ? creep

Exit: (9) male(andrew) ? creep

Call: (9) elder\_than(andrew, \_798) ? creep

Exit: (9) elder\_than(andrew, edward) ? creep

Exit: (8) succession(andrew) ? creep

X = andrew ;

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(edward, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) male(edward) ? creep

Exit: (9) male(edward) ? creep

Call: (9) elder\_than(edward, \_798) ? creep

Fail: (9) elder\_than(edward, \_798) ? creep

Redo: (8) succession(\_582) ? creep

Call: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(charles, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) male(charles) ? creep

Exit: (9) male(charles) ? creep

^ Call: (9) not(elder\_than(charles, \_784)) ? creep

Call: (10) elder\_than(charles, \_784) ? creep

Exit: (10) elder\_than(charles, ann) ? creep

^ Fail: (9) not(user:elder\_than(charles, \_784)) ? creep

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(ann, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) male(ann) ? creep

Fail: (9) male(ann) ? creep

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(andrew, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) male(andrew) ? creep

Exit: (9) male(andrew) ? creep

^ Call: (9) not(elder\_than(andrew, \_784)) ? creep

Call: (10) elder\_than(andrew, \_784) ? creep

Exit: (10) elder\_than(andrew, edward) ? creep

^ Fail: (9) not(user:elder\_than(andrew, \_784)) ? creep

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(edward, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) male(edward) ? creep

Exit: (9) male(edward) ? creep

^ Call: (9) not(elder\_than(edward, \_784)) ? creep

Call: (10) elder\_than(edward, \_784) ? creep

Fail: (10) elder\_than(edward, \_784) ? creep

^ Exit: (9) not(user:elder\_than(edward, \_784)) ? creep

Exit: (8) succession(edward) ? creep

X = edward ;

Redo: (8) succession(\_582) ? creep

Call: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(charles, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) female(charles) ? creep

Fail: (9) female(charles) ? creep

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(ann, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) female(ann) ? creep

Exit: (9) female(ann) ? creep

Call: (9) elder\_than(ann, \_798) ? creep

Exit: (9) elder\_than(ann, andrew) ? creep

Exit: (8) succession(ann) ? creep

X = ann ;

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(andrew, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) female(andrew) ? creep

Fail: (9) female(andrew) ? creep

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(edward, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) female(edward) ? creep

Fail: (9) female(edward) ? creep

Redo: (8) succession(\_582) ? creep

Call: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(charles, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) female(charles) ? creep Fail:

(9) female(charles) ? creep

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(ann, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) female(ann) ? creep

Exit: (9) female(ann) ? creep

^ Call: (9) not(elder\_than(ann, \_784)) ? creep

Call: (10) elder\_than(ann, \_784) ? creep

Exit: (10) elder\_than(ann, andrew) ? creep

^ Fail: (9) not(user:elder\_than(ann, \_784)) ? creep

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(andrew, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) female(andrew) ? creep

Fail: (9) female(andrew) ? creep

Redo: (9) offspring(\_582, \_798) ? creep

Exit: (9) offspring(edward, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) female(edward) ? creep

Fail: (9) female(edward) ? creep

Fail: (8) succession(\_582) ? creep

false.

1. Knowledge Base:

male(charles).

male(andrew).

male(edward).

female(ann).

female(elizabeth).

monarch(elizabeth).

queen(elizabeth).

offspring(charles,elizabeth).

offspring(ann,elizabeth).

offspring(andrew,elizabeth).

offspring(edward,elizabeth).

elder\_than(charles,ann).

elder\_than(ann,andrew).

elder\_than(andrew,edward).

succession(X) :- offspring(X,Z),monarch(Z),elder\_than(X,Y).

succession(X) :- offspring(X,Z),monarch(Z),not(elder\_than(X,Y)).

Answer –

[debug] ?- succession(X).

X = charles ;

X = ann ;

X = andrew ;

X = edward.

Trace –

Ѐ[trace] ?- succession(X).

Call: (8) succession(\_2102) ? creep

Call: (9) offspring(\_2102, \_2314) ? creep

Exit: (9) offspring(charles, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) elder\_than(charles, \_2314) ? creep

Exit: (9) elder\_than(charles, ann) ? creep

Exit: (8) succession(charles) ? creep

X = charles ;

Redo: (9) offspring(\_2102, \_2314) ? creep

Exit: (9) offspring(ann, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) elder\_than(ann, \_2314) ? creep

Exit: (9) elder\_than(ann, andrew) ? creep

Exit: (8) succession(ann) ? creep

X = ann ;

Redo: (9) offspring(\_2102, \_2314) ? creep

Exit: (9) offspring(andrew, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) elder\_than(andrew, \_2314) ? creep

Exit: (9) elder\_than(andrew, edward) ? creep

Exit: (8) succession(andrew) ? creep

X = andrew ;

Redo: (9) offspring(\_582, \_794) ? creep

Exit: (9) offspring(edward, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

Call: (9) elder\_than(edward, \_794) ? creep

Fail: (9) elder\_than(edward, \_794) ? creep

Redo: (8) succession(\_582) ? creep

Call: (9) offspring(\_582, \_794) ? creep

Exit: (9) offspring(charles, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

^ Call: (9) not(elder\_than(charles, \_780)) ? creep

Call: (10) elder\_than(charles, \_780) ? creep

Exit: (10) elder\_than(charles, ann) ? creep

^ Fail: (9) not(user:elder\_than(charles, \_780)) ? creep

Redo: (9) offspring(\_582, \_794) ? creep

Exit: (9) offspring(ann, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

^ Call: (9) not(elder\_than(ann, \_780)) ? creep

Call: (10) elder\_than(ann, \_780) ? creep

Exit: (10) elder\_than(ann, andrew) ? creep

^ Fail: (9) not(user:elder\_than(ann, \_780)) ? creep

Redo: (9) offspring(\_582, \_794) ? creep

Exit: (9) offspring(andrew, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

^ Call: (9) not(elder\_than(andrew, \_780)) ? creep

Call: (10) elder\_than(andrew, \_780) ? creep

Exit: (10) elder\_than(andrew, edward) ? creep

^ Fail: (9) not(user:elder\_than(andrew, \_780)) ? creep

Redo: (9) offspring(\_582, \_794) ? creep

Exit: (9) offspring(edward, elizabeth) ? creep

Call: (9) monarch(elizabeth) ? creep

Exit: (9) monarch(elizabeth) ? creep

^ Call: (9) not(elder\_than(edward, \_780)) ? creep

Call: (10) elder\_than(edward, \_780) ? creep

Fail: (10) elder\_than(edward, \_780) ? creep

^ Exit: (9) not(user:elder\_than(edward, \_780)) ? creep

Exit: (8) succession(edward) ? creep

X = edward.

|  |
| --- |
|  |