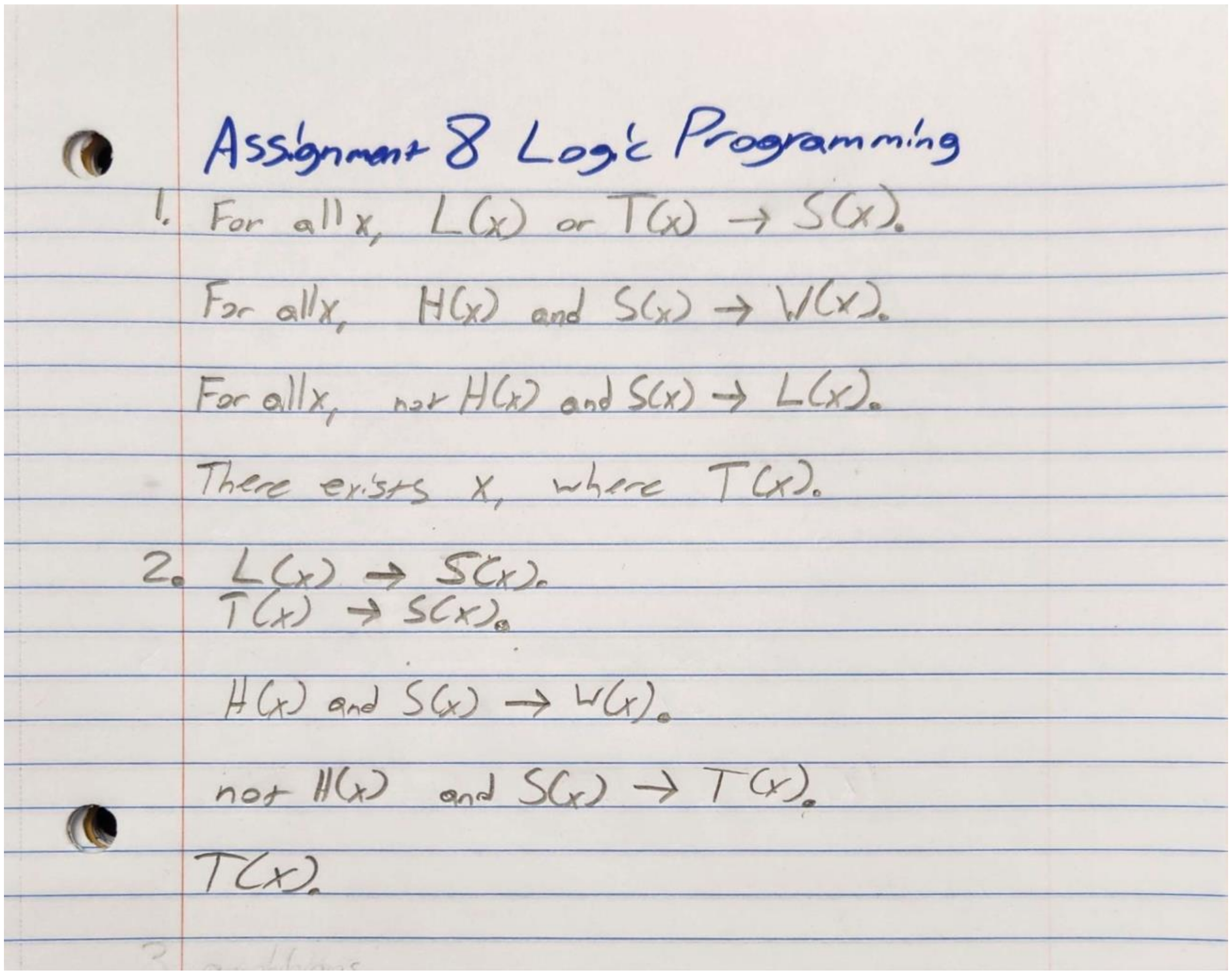
Hayden Coble
hcoble@my.athens.edu
CS 414

Assignment 8: Logic Programming



Assignment 8 Logic Programming

1. For all x, L(x) or T(x) $\to$ S(x).

   For all x, H(x) and S(x) $\to$ W(x).

   For all x, not H(x) and S(x) $\to$ L(x).

   There exists x, where T(x).

2. L(x) $\to$ S(x).
   T(x) $\to$ S(x).

   H(x) and S(x) $\to$ W(x).

   not H(x) and S(x) $\to$ T(x).

   T(x).

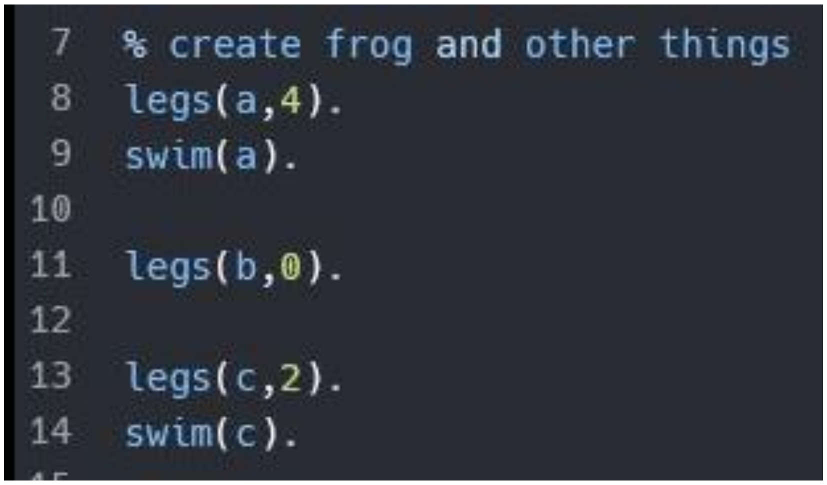
3. Prolog Code:

```
amphibian(X):- legs(X,4), swim(X); legs(X,0), swim(X) \= true.

frog(X):- amphibian(X),legs(X,4),swim(X).
```

4. See second line in text box above.

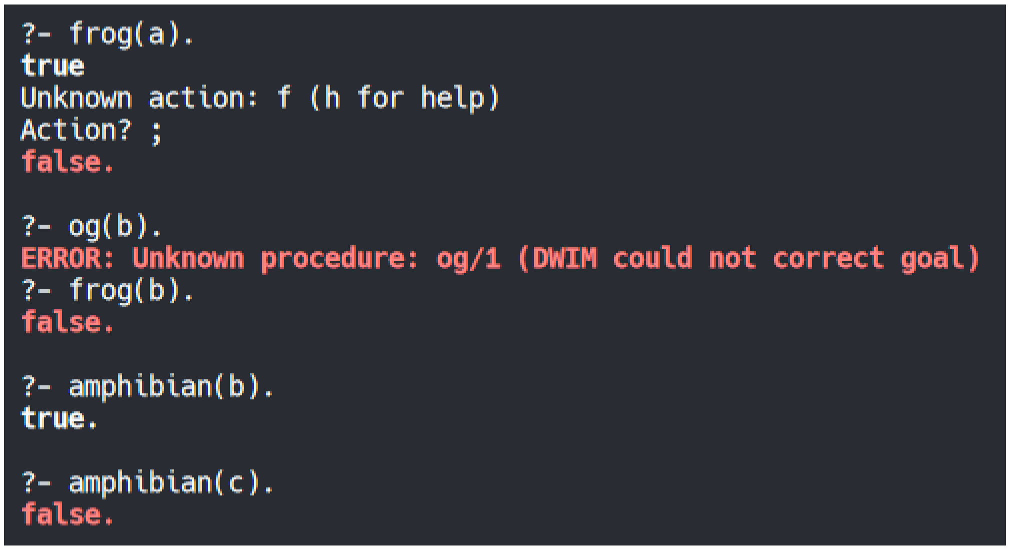
Hayden Coble
hcoble@my.athens.edu
CS 414

5. I defined the following facts about a, b, and c…



```
% create frog and other things
legs(a,4).
swim(a).

legs(b,0).

legs(c,2).
swim(c).
```

…then ran these queries:

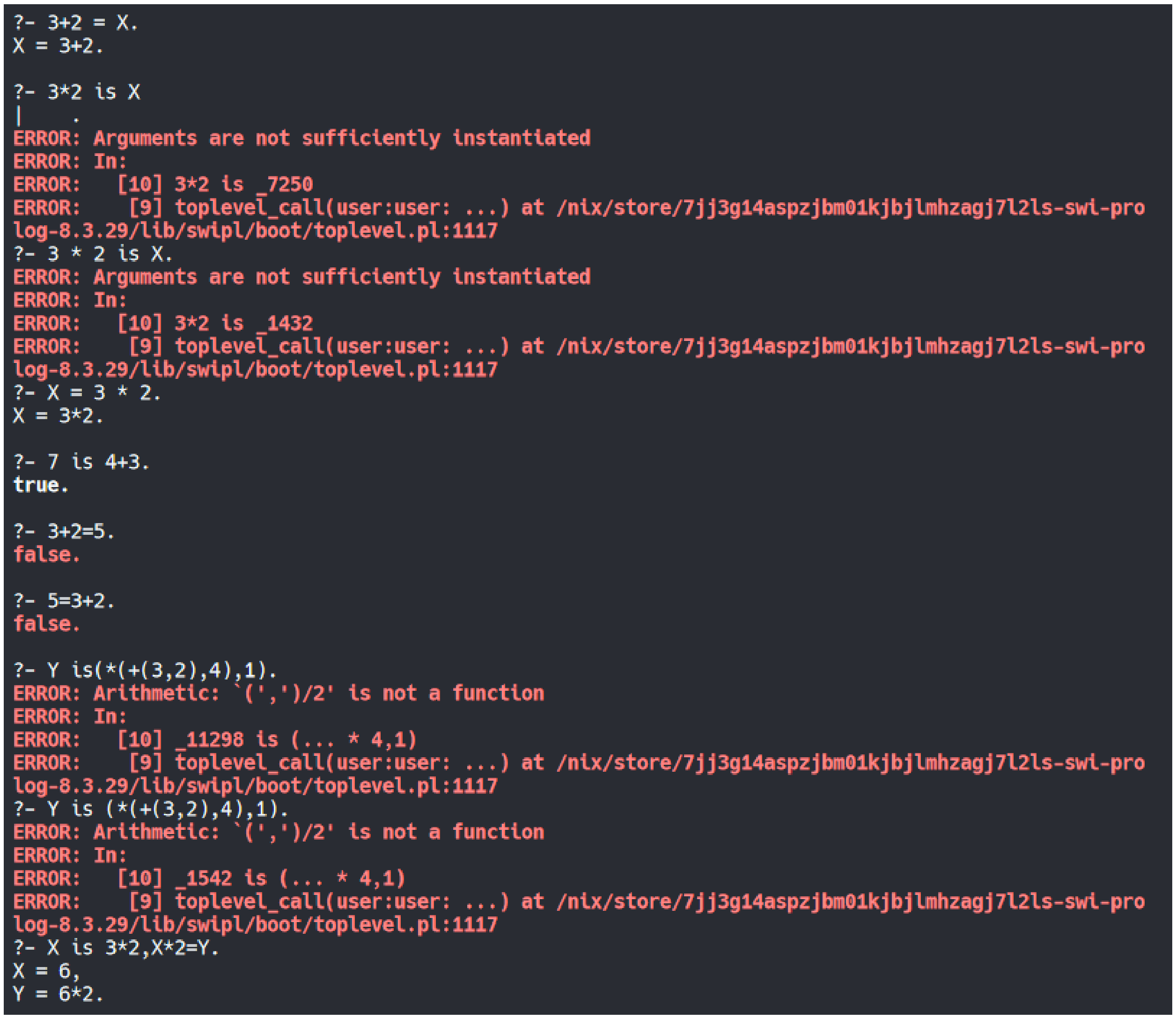


```
?- frog(a).
true
Unknown action: f (h for help)
Action? ;
false.

?- og(b).
ERROR: Unknown procedure: og/1 (DWIM could not correct goal)
?- frog(b).
false.

?- amphibian(b).
true.

?- amphibian(c).
false.
```

Knowing the facts, I can verify whether my rules are enforced properly. A is a frog, b isn't but is an amphibian, and c is neither; they are correctly enforced.

Hayden Coble
hcoble@my.athens.edu
CS 414

6. I get the following results:



```
?- 3+2 = X.
X = 3+2.

?- 3*2 is X
|    .
ERROR: Arguments are not sufficiently instantiated
ERROR: In:
ERROR:   [10] 3*2 is _7250
ERROR:    [9] toplevel_call(user:user: ...) at /nix/store/7jj3g14aspzjbm01kjbjlmhzagj7l2ls-swi-pro
log-8.3.29/lib/swipl/boot/toplevel.pl:1117
?- 3 * 2 is X.
ERROR: Arguments are not sufficiently instantiated
ERROR: In:
ERROR:   [10] 3*2 is _1432
ERROR:    [9] toplevel_call(user:user: ...) at /nix/store/7jj3g14aspzjbm01kjbjlmhzagj7l2ls-swi-pro
log-8.3.29/lib/swipl/boot/toplevel.pl:1117
?- X = 3 * 2.
X = 3*2.

?- 7 is 4+3.
true.

?- 3+2=5.
false.

?- 5=3+2.
false.

?- Y is(*(+(3,2),4),1).
ERROR: Arithmetic: `(',')/2' is not a function
ERROR: In:
ERROR:   [10] _11298 is (... * 4,1)
ERROR:    [9] toplevel_call(user:user: ...) at /nix/store/7jj3g14aspzjbm01kjbjlmhzagj7l2ls-swi-pro
log-8.3.29/lib/swipl/boot/toplevel.pl:1117
?- Y is (*(+(3,2),4),1).
ERROR: Arithmetic: `(',')/2' is not a function
ERROR: In:
ERROR:   [10] _1542 is (... * 4,1)
ERROR:    [9] toplevel_call(user:user: ...) at /nix/store/7jj3g14aspzjbm01kjbjlmhzagj7l2ls-swi-pro
log-8.3.29/lib/swipl/boot/toplevel.pl:1117
?- X is 3*2,X*2=Y.
X = 6,
Y = 6*2.
```

a. X is undefined, so there is no issue saying that X = 3+2. 3+2 is not evaluated, and this satisfies the logic. X isn't 5, it's 3+2, which = 3+2, so X=3+2 is returned.
b. 3*2 is X results in an error. "is" takes an expression on the right-hand side and a number on the left-hand side. 3*2 is not a number, hence the error.
c. X = 3*2 assigns 3*2 to X; it isn't evaluated; a statement showing X = 3*2 is returned.
d. 7 is 4+3 is true. "is" evaluates each side to check if they are the same, thus this returns true.
e. 3+2 = 5 is false because 3+2 is not the same as 5. When evaluated, they are both equivalent to 5, but can not be unified because they are not the same type.
f. Y is (*(+(3,2),4),1) yields an error. An operator is missing in this statement. The compiler understands that there should be an operator between the left and right arguments (or before outside the parenthesis), and thinks we intended to use the comma as an operator. To support this, changing the last comma results in Y = 21 due to the mixing of notations.

Hayden Coble
hcoble@my.athens.edu
CS 414

g. X is 3*2, X*2=Y:
X = 6 because "is" implies X is a number equal to what 3*2 evaluates to.
, is the logical and operator.
X*2 = Y implies Y=6*2. This is because the = operator unifies the two arguments without evaluating them (X is already evaluated from the first half). Y is the expression of X's value * 2, thus Y=6*2.