Jorge Adrian Padilla Velasco
A00570894
october 18th 2018

Arithmetic in Prolog

Exercise 1

How does Prolog respond to the following queries?

1. **X = 3 * 4.**
   X = 3*4.

2. **X is 3 * 4.**
   X = 12.

3. **4 is X.**
   ERROR: is/2: Arguments are not sufficiently instantiated

4. **X = Y.**
   X = Y.

5. **3 is 1 + 2.**
   true.

6. **3 is + (1 , 2).**
   true.

7. **3 is X + 2.**
   ERROR: is/2: Arguments are not sufficiently instantiated

8. **X is 1+ 2.**
   X = 3.

9. **1 + 2 is 1 + 2.**
   false.

10. **is(X, + (1, 2)).**
    X = 3.

11. **3 + 2 = + (3, 2).**
    true.

12. ***(7, 5) = 7 * 5.**
    true.

13. ***(7, + (3, 2)) = 7 * (3 + 2).**
    true.

14.*(7, (3 + 2)) = 7 * (3 + 2).
true.

15.7 * 3 + 2 = *(7, + (3, 2)).
false.

16.*(7, (3 + 2)) = 7 * (+ (3, 2)).

true.

Exercise 2

1. Define a 2-place predicate increment that holds only when its second argument is an integer one larger than its first argument. For example, increment(4,5) should hold, but increment(4,6) should not.

   ```
   increment(X,Y) :- Y is X + 1.
   ```

2. Define a 3-place predicate sum that holds only when its third argument is the sum of the first two arguments. For example, sum(4,5,9) should hold, but sum(4,6,12) should not.

   ```
   sum(X,Y,Z) :- Z is (X + Y).
   ```

Exercise 3

Write a predicate addone/2 whose first argument is a list of integers, and whose second argument is the list of integers obtained by adding 1 to each integer in the first list. For example, the query

?- addone([1,2,7,2],X).

should give

X = [2,3,8,3].

```
addone([], []).
addone([H1 | T1], [H2 | T2]) :- is(H2, +(H1, 1)), addone(T1, T2).
```