Donovan Roseau
November 3, 2016
CSCI 320

CSCI 320: Assignment 3

1a) north_of(tokyo,chicago).

Call: north_of(tokyo, chicago) ?

Call: location(tokyo, _1, _2) ?

Exit: location(tokyo, 35, 141) ?

Call: location(chicago, _1, _2) ?

Exit: location(chicago, 42, 88) ?

Call: 42<35 ?

Fail: 42<35 ?

Fail: north_of(tokyo, chicago) ?

false.

1b) north_of(X,new_york).

Call: north_of(_0, new_york) ?

Call: location(_0, _1, _2) ?

Exit: location(new_york, 41, 74) ?

Call: location(new_york, _1, _2) ?

Exit: location(new_york, 41, 74) ?

Call: 41<41 ?

Fail: 41<41 ?

Redo: location(_0, _1, _2) ?

Exit: location(chicago, 42, 88) ?

Call: location(new_york, _1, _2) ?

Exit: location(new_york, 41, 74) ?

Call: 41<42 ?

Exit: 41<42 ?

Exit: north_of(chicago, new_york) ?

X = Chicago;

Redo: location(_0, _1, _2) ?

Exit: location(tokyo, 35, 141) ?

Call: location(new_york, _1, _2) ?

Exit: location(new_york, 41, 74) ?

Call: 41<35 ?

Fail: 41<35 ?

Redo: location(_0, _1, _2) ?

Exit: location(oslo, 60, 11) ?

Call: location(new_york, _1, _2) ?

Exit: location(new_york, 41, 74) ?

Call: 41<60 ?

Exit: 41<60 ?

Exit: north_of(oslo, new_york) ?

X = oslo ;

Redo: location(_0, _1, _2) ?

Exit: location(quito, 0, 80) ?

Call: location(new_york, _1, _2) ?

Exit: location(new_york, 41, 74) ?

Call: 41<0 ?

Fail: 41<0 ?

Redo: location(_0, _1, _2) ?

Exit: location(cairo, 30, 30) ?

Call: location(new_york, _1, _2) ?

Exit: location(new_york, 41, 74)?

Call: 41 < 30?

Fail: 41<30?

Fail: north_of(_0, new_york)?

False.

Donovan Roseau
November 3, 2016
CSCI 320

2) perennial_garden(X).

Call: perennial_garden(_0) ?

Call: flower(_0) ?

Exit: flower(phlox) ?

Call: type(phlox, perennial) ?

Exit: type(phlox, perennial) ?

Exit: perennial_garden(phlox) ?

X = phlox ;

Redo: flower(_0) ?

Exit: flower(petunia) ?

Call: type(petunia, perennial) ?

Fail: type(petunia, perennial) ?

Redo: flower(_0) ?

Exit: flower(rose) ?

Call: type(rose, perennial) ?

Fail: type(rose, perennial) ?

Redo: flower(_0) ?

Exit: flower(daisy) ?

Call: type(daisy, perennial) ?

Exit: type(daisy, perennial) ?

Exit: perennial_garden(daisy) ?

X = daisy ;

Redo: type(daisy, perennial) ?

Fail: type(daisy, perennial) ?

Fail: perennial_garden(_0) ?

false.

Donovan Roseau
November 3, 2016
CSCI 320

3) ?- reverse([1,2],Z).

Call: reverse([1, 2], _1) ?

Call: reverse([2], _1) ?

Call: reverse([], _1) ?

Exit: reverse([], []) ?

Call: lists:append([], [2], _2) ?

Exit: lists:append([], [2], [2]) ?

Exit: reverse([2], [2]) ?

Call: lists:append([2], [1], _2) ?

Exit: lists:append([2], [1], [2, 1]) ?

Exit: reverse([1, 2], [2, 1]) ?

Z = [2, 1]

4) reverse([1,2],Z).

Call: reverse([1, 2], _1) ?

Call: reverse([2], _1) ?

Call: reverse([], _1) ?

Exit: reverse([], []) ?

Call: lists:append([], [2], _1) ?

Exit: lists:append([], [2], [2]) ?

Exit: reverse([2], [2]) ?

Call: lists:append([2], [1], _1) ?

Exit: lists:append([2], [1], [2, 1]) ?

Exit: reverse([1, 2], [2, 1]) ?

Z = [2, 1].

5) Fibonacci Sequence

```
fib(0,0).

fib(1,1).

fib(N,FN) :-    N > 1,
                X is N - 1, Y is N - 2,
```

fib(X,A), fib(Y,B), plus(A,B,FN).