**Project 4**

**Regan O’Donnell**

**COP4020**

Start situation – implemented

move – implemented

block – implemented

location – implemented

move 2 – implemented

move 3 – implemented

clear\_top – implemented

move 4 – implemented

clear (recursive) – implemented

Here are some predicates you can try with my program, they can also be found commented at the bottom of my submission:

% Test Start Situation

% ?- start(S).

% Test Move Predicate

% ?- start(S), move(b2, surface, S, NewSituation).

% Test Block and Location Predicates

% ?- start(S), block(Block, S).

% ?- start(S), block(b3, S).

% ?- start(S), block(surface, S).

% ?- start(S), location(Location, S).

% % Test Move2 Predicate

% ?- start(S), move2(b2, surface, S, NewSituation).

% ?- start(S), move2(surface, b1, S, \_).

% ?- start(S), move2(b1, table, S, \_).

% % Test Clear\_Top Predicate

% ?- start(S), clear\_top(surface, S).

% ?- start(S), clear\_top(b1, S).

% ?- start(S), clear\_top(b2, S).

% % Test Move3 Predicate

% ?- start(S), move3(b2, surface, S, \_).

% ?- start(S), move3(surface, b1, S, \_).

% % Test Clear Predicate

% ?- start(S), clear(b2, S, \_).

% ?- start(S), clear(b1, S, \_).

% % Test Recursive Clear

% start2([on(b1, surface), on(b2, b1), on(b3, b2)]).

% ?- start2(S), clear(b1, S, \_).