

```

function [L, P] = lqr_infinite_horizon_solution(Q, R)

%% find the infinite horizon L and P through running LQR back-ups
%% until norm(L_new - L_current, 2) <= 1e-4
dt = 0.1;
mc = 10; mp = 2.; l = 1.; g = 9.81;

% TODO write A,B matrices
a1 = mp*g/mc;
a2 = (mc+mp)*g/(l*mc);

A = [0, 0, 1, 0;
     0, 0, 0, 1;
     0, a1, 0, 0;
     0, a2, 0, 0];

B = [0; 0; 1/mc; 1/(l*mc)];

%Riccati recursion until approximate convergence
N = 1;
P_1 = RiccatiRecursion(A,B,Q,R,N);
L_current = pinv(R + B'*P_1*B)*(B'*P_1*A);

N = 2;
P_2 = RiccatiRecursion(A,B,Q,R,N);
L_new = pinv(R + B'*P_2*B)*(B'*P_2*A);

P = P_2;

while norm((L_new-L_current),2) > 1e-4
    L_current = L_new;
    N = N+1;
    P = RiccatiRecursion(A,B,Q,R,N);
    L_new = pinv(R + B'*P*B)*(B'*P*A);
end
L = L_new;

end

function P = RiccatiRecursion(A,B,Q,R,N)
if N == 0
    P = Q;
else
    P_k = RiccatiRecursion(A,B,Q,R,N-1);
    P_kprev = A'*P_k*A-(A'*P_k*B)*pinv(R+B'*P_k*B)*(B'*P_k*A)+Q;
    P = P_kprev;
end

```

end