

```
% Exercise 4.5 – modifications on the jacks car rental example.
%
% See ePage 262 in the Sutton book.
%
% Written by:
% --
% John L. Weatherwax                2007-12-03
%
% email: wax@alum.mit.edu
%
% Please send comments and especially bug reports to the
% above email address.
%
%-----

clc;
close all;

gamma = 0.9;

% the maximum number of cars we can store (overnight) at each site:
max_n_cars = 20;
%max_n_cars = 10;
%max_n_cars = 5;

max_num_cars_can_transfer = 5;

% the maximum number of cars we can store at a sight without having to pay the $4 parking
cost.
max_cars_can_store = 10;
%max_cars_can_store = 5;

% the parameters of the environment:
if( 0 )
    lambda_A_return = 3; % a debugging case:
    lambda_A_rental = 3;
    lambda_B_return = 3;
    lambda_B_rental = 3;
else
    lambda_A_return = 3;
    lambda_A_rental = 3;
    lambda_B_return = 2;
    lambda_B_rental = 4;
end

% precompute the rewards and transition probabilities:
[Ra,Pa] =
cmpt_P_and_R(lambda_A_rental,lambda_A_return,max_n_cars,max_num_cars_can_transfer);
[Rb,Pb] =
cmpt_P_and_R(lambda_B_rental,lambda_B_return,max_n_cars,max_num_cars_can_transfer);

% initial state value function:
V = zeros(max_n_cars+1,max_n_cars+1);

% initial policy (now includes employee transfer) :
pol_pi = zeros(max_n_cars+1,max_n_cars+1); emp_pol_pi = zeros(max_n_cars+1,max_n_cars+1);

policyStable = 0; iterNum = 0;
while( ~policyStable )
```

```

% plot the current policy:
if( iterNum~=0 )
    %if( 0 )
        figure; subplot( 1,2,1 );
        imagesc( 0:max_n_cars, 0:max_n_cars, pol_pi ); colorbar; xlabel( 'num at B' );
ylabel( 'num at A' ); axis xy;
        %title( ['current policy iter=', num2str(iterNum)] );
        subplot( 1,2,2 );
        imagesc( 0:max_n_cars, 0:max_n_cars, emp_pol_pi ); colorbar; xlabel( 'num at B' );
ylabel( 'num at A' ); axis xy;
        drawnow;
        fn=sprintf('policy_iter_both_%d.eps',iterNum); saveas( gcf, fn, 'eps2' );
    %else
        figure;
        imagesc( 0:max_n_cars, 0:max_n_cars, pol_pi+emp_pol_pi ); colorbar; xlabel( 'num at
B' ); ylabel( 'num at A' ); axis xy;
        %title( ['current policy iter=', num2str(iterNum)] );
        drawnow;
        fn=sprintf('policy_iter_combined_%d.eps',iterNum); saveas( gcf, fn, 'eps2' );
    %end
end

% evaluate the state-value function under this policy:
V =
ex_4_5_policy_evaluation(V,pol_pi,emp_pol_pi,gamma,Ra,Pa,Rb,Pb,max_num_cars_can_transfer,
max_cars_can_store);
if( 1 )
    figure; imagesc( 0:max_n_cars, 0:max_n_cars, V ); colorbar;
    xlabel( 'num at B' ); ylabel( 'num at A' );
    title( ['current state-value function iter=', num2str(iterNum)] ); axis xy; drawnow;
    fn=sprintf('state_value_fn_iter_%d.eps',iterNum); saveas( gcf, fn, 'eps2' );
end

% compute an improved policy using the most recent as a base:
[pol_pi,emp_pol_pi,policyStable] =
ex_4_5_policy_improvement(pol_pi,emp_pol_pi,V,gamma,Ra,Pa,Rb,Pb,max_num_cars_can_transfer
,max_cars_can_store);

    iterNum=iterNum+1;
    %if( iterNum==2 ) break; end;
end
% plot the current policy:
%figure; imagesc( pol_pi ); colorbar; title( ['current policy iter=', num2str(iterNum)]
); drawnow;

```