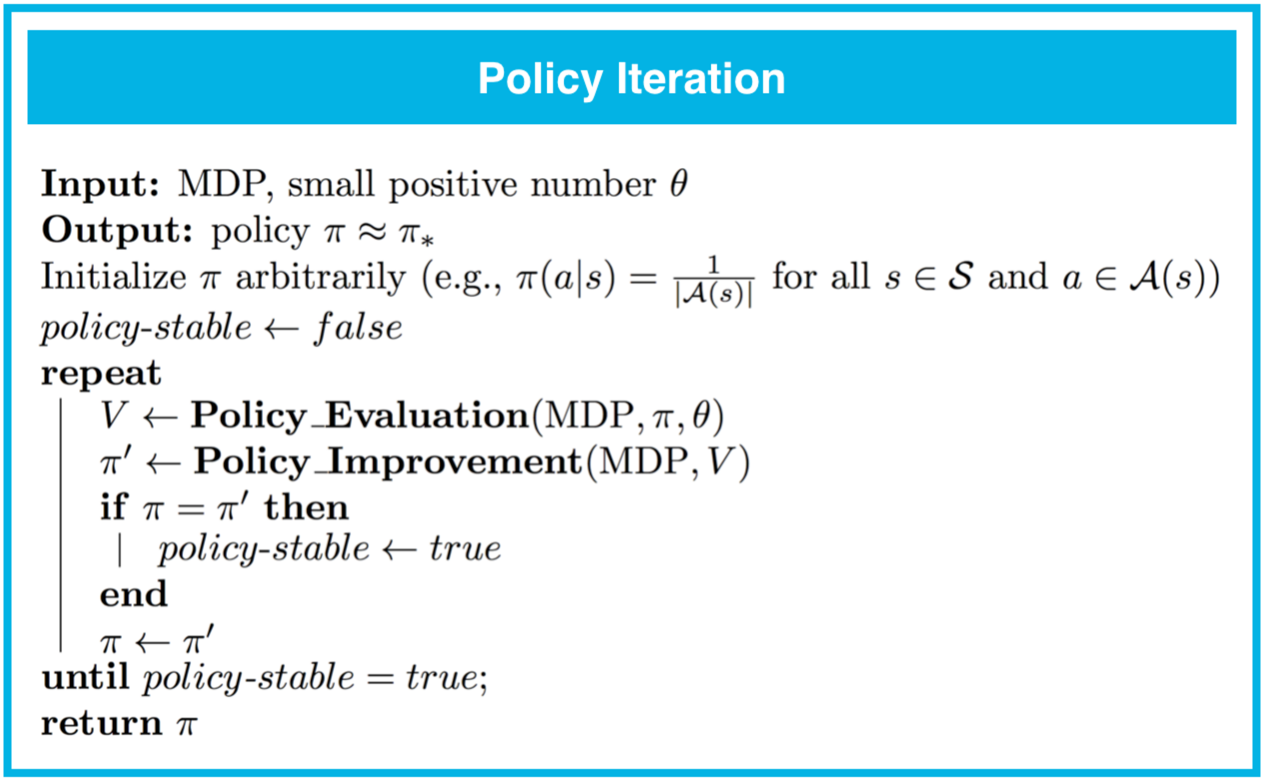
Implementation: Policy Iteration

In the previous concept, you learned about **policy iteration**, which proceeds as a series of alternating policy evaluation and improvement steps. Policy iteration is guaranteed to find the optimal policy for any finite Markov decision process (MDP) in a finite number of iterations. The pseudocode can be found below.



Policy Iteration

```
Input: MDP, small positive number $\theta$
Output: policy $\pi \approx \pi_*$
Initialize $\pi$ arbitrarily (e.g., $\pi(a|s) = \frac{1}{|\mathcal{A}(s)|}$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}(s)$)
$policy\text{-}stable \leftarrow false$
repeat
    $V \leftarrow$ Policy_Evaluation(MDP, $\pi$, $\theta$)
    $\pi' \leftarrow$ Policy_Improvement(MDP, $V$)
    if $\pi = \pi'$ then
        $policy\text{-}stable \leftarrow true$
    end
    $\pi \leftarrow \pi'$
until $policy\text{-}stable = true$;
return $\pi$
```

Please use the next concept to complete **Part 4: Policy Iteration** of `Dynamic_Programming.ipynb` . Remember to save your work!

If you'd like to reference the pseudocode while working on the notebook, you are encouraged to open this sheet in a new window.

Feel free to check your solution by looking at the corresponding section in `Dynamic_Programming_Solution.ipynb` .

NEXT

Implementation