Aprendizagem por Reforço

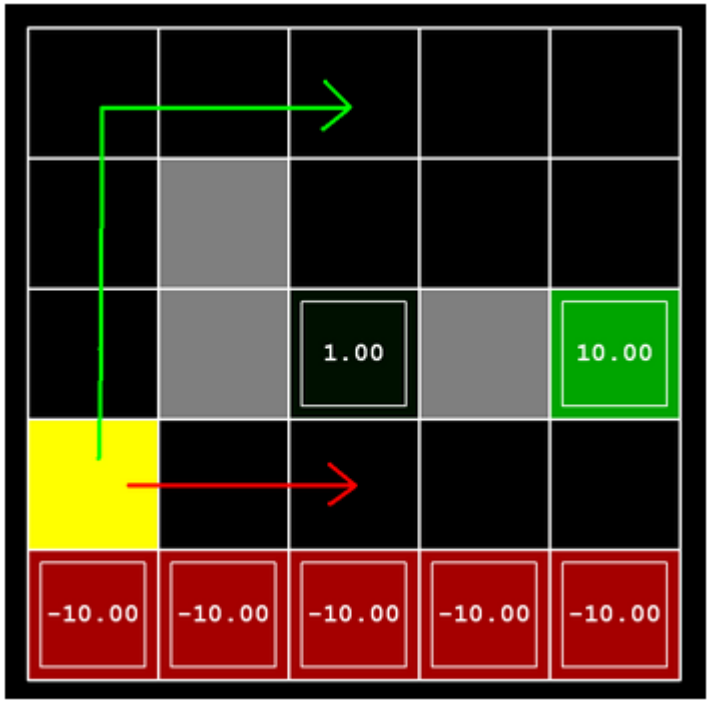
Trabalho de Implementação
Disciplina de Inteligência Artificial II
Prof Carine Webber

Tema do Trabalho

- Aprendizagem por Reforço
  - Estudo do algoritmo Q-Learning
  - Modelagem do problema de alcançar um estado objetivo
  - Implementação do algoritmo Q-Learning para resolver o problema definido

Aprendizagem sem Supervisor

- Suponha que um agente seja colocado em um ambiente e tenha que aprender a se comportar com sucesso nesse ambiente.

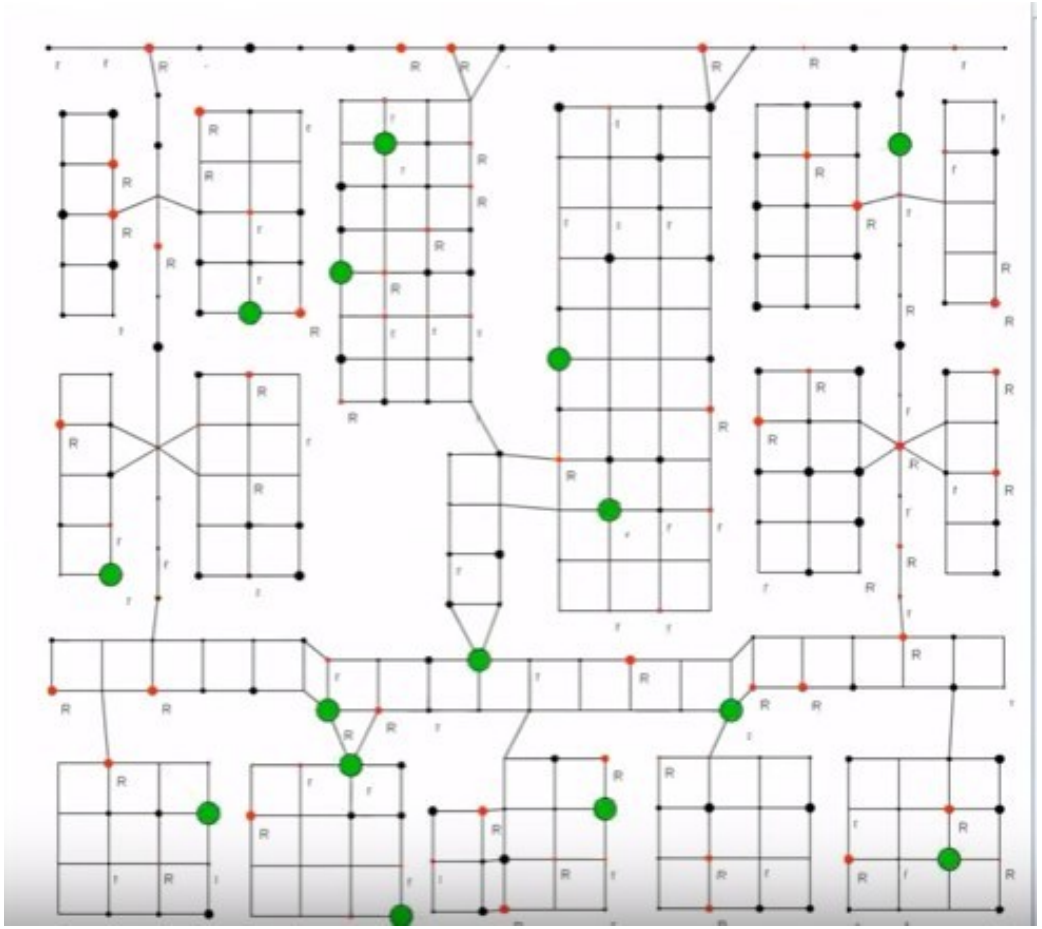


Exemplo

	Estado	Ação	Recompensa
Agente patrulhador	Posição no mapa (atual e passadas), ociosidade da vizinhança, etc...	Ir para algum lugar vizinho do mapa	Ociosidade (tempo sem visitas) do lugar visitado atualmente

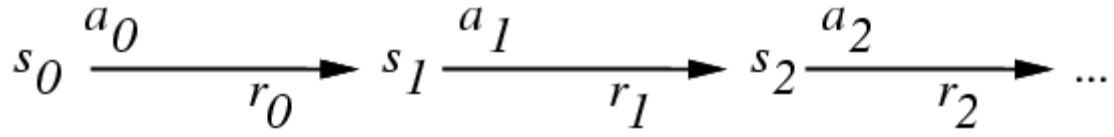
Exemplo

- https://www.youtube.com/watch?v=hrWnqJeQOQI



Conceitos Básicos

- Processo de decisão de Markov (MDP)
  - Conjunto de estados **S**
  - Conjunto de ações **A**
  - Uma função de recompensa **r(s,a)**
  - Uma função de transição de estados (pode ser estocástica) δ**(s,a)**
- Política de ações π**(s)** :
  - π: **S** $\Rightarrow$ **A**



Função de Recompensa

- Feedback do ambiente sobre o comportamento do agente
- Indicada por $r:(S \times A) \rightarrow R$
  - r(s,a) indica a recompensa recebida quando se está no estado **s** e se executa a ação **a**
  - Pode ser determinística ou estocástica

Transição de Estados

- δ: (S x A) -> S
- δ(s,a) indica em qual estado o agente está, dado que:
  - estava no estado s
  - executou a ação a

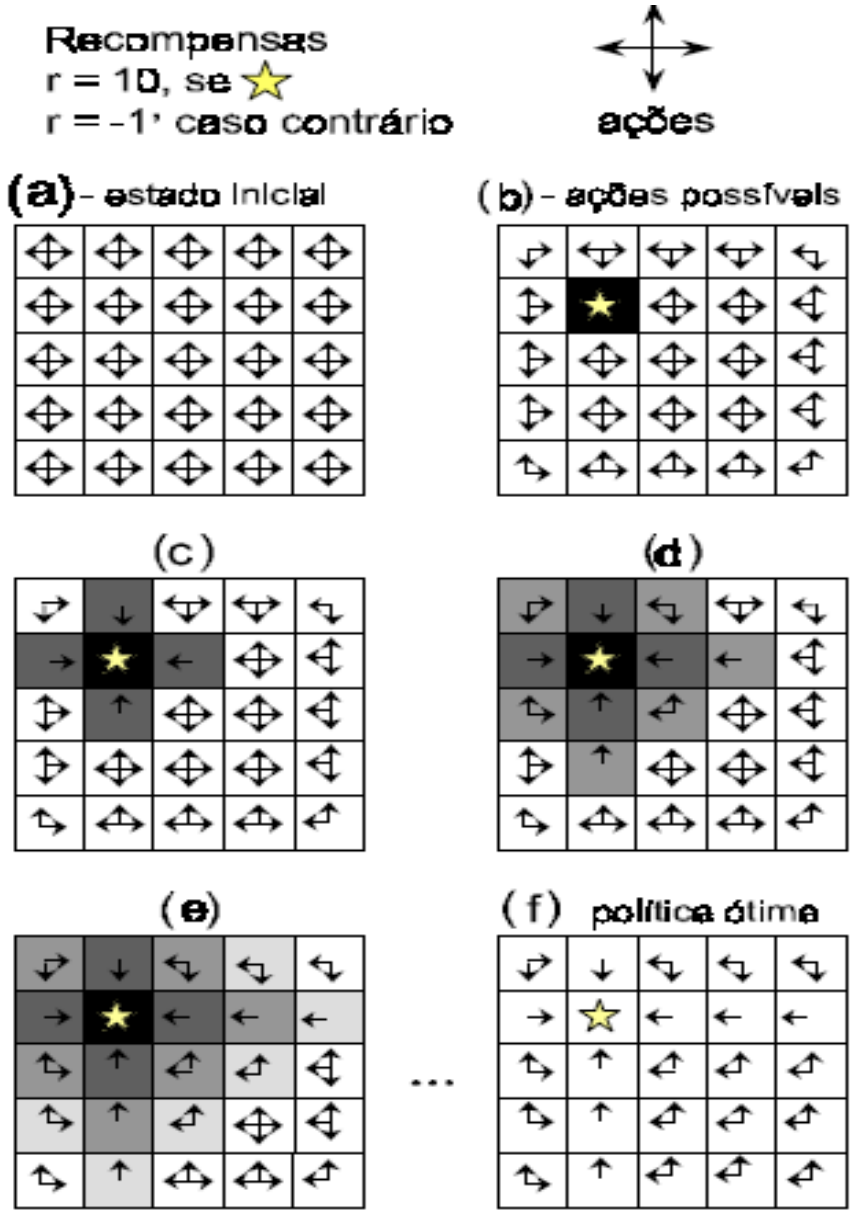
Política de ações (π)

- Função que modela o comportamento do agente
- Mapeia estados em ações
- Pode ser vista como um conjunto de regras do tipo:
  - $s_n \rightarrow a_m$
- Exemplo:
  - Se estado s = (predador próximo, arma sem munição e tempo acabando) então

    ação a = (usar magia);
  - Se estado s =(predador próximo, arma com munição) então

    ação a = (disparar 1 tiro)

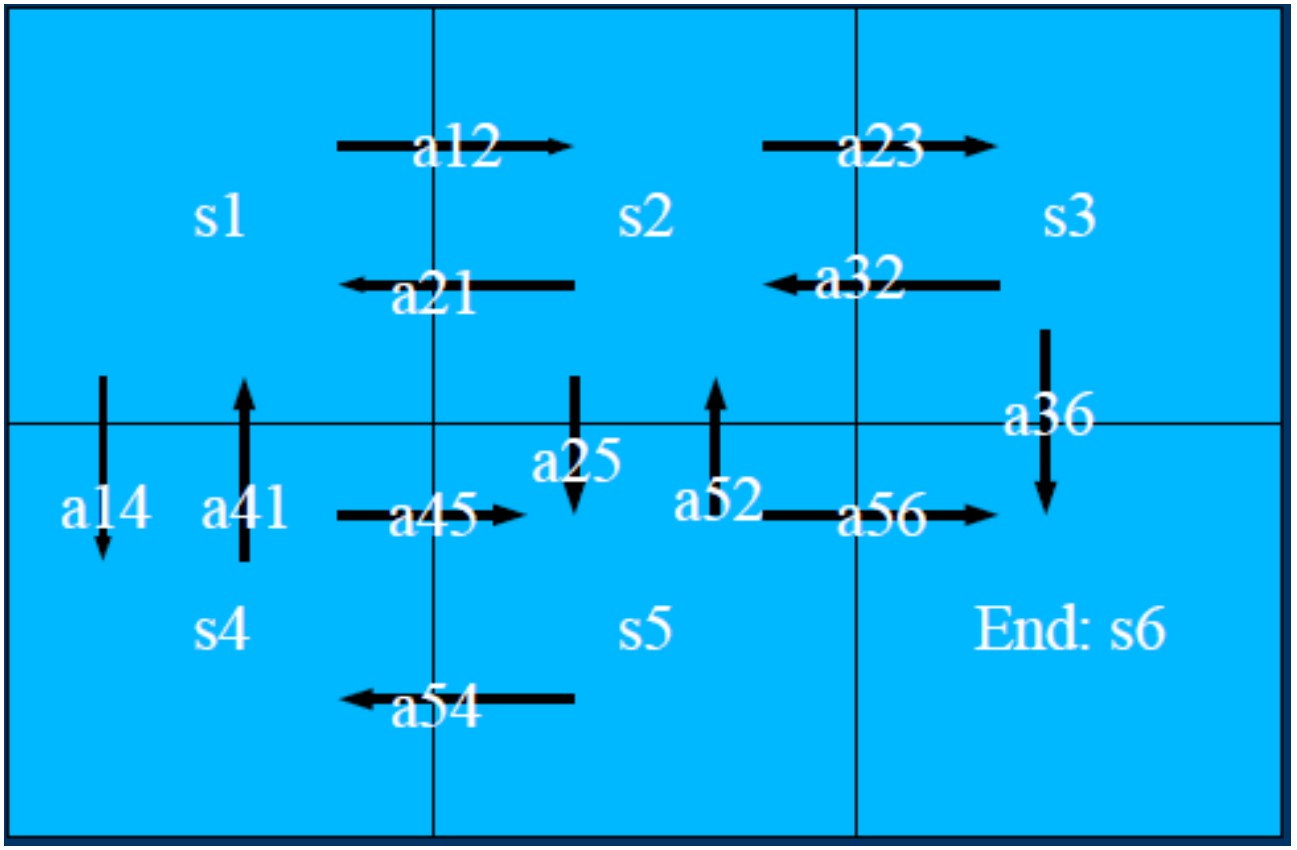
Exemplo de Construção de Política Ótima



Algoritmo Q-Learning

1. Inicialize Q(s,a) arbitrariamente
2. Repita (para cada episódio)
   2.1 Inicialize `s`
   2.2 Repita para cada passo do episódio
      2.2.1 Escolha α ϵ A(s)
      2.2.2. Execute a ação α
      2.2.3. Observe os valores s' e r
      2.2.4. $Q(s,a)=r(s,a)+\gamma max_{a'}(Q(s',a'))$
      2.2.5. s<- s'
   2.3.até que s seja terminal

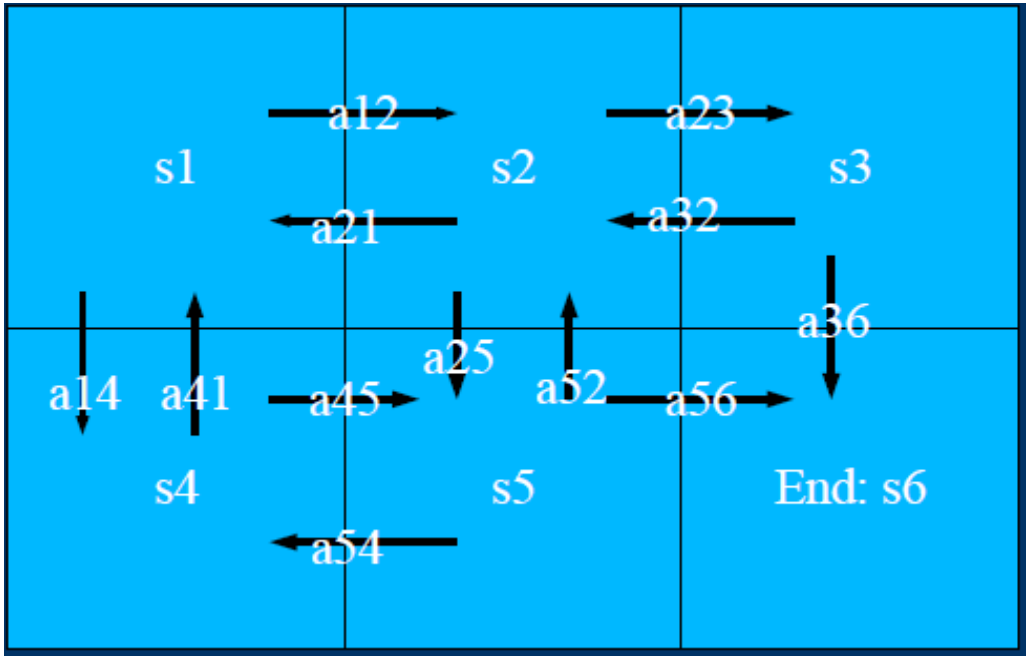
Exemplo – *Alcançar o destino S6*



$\gamma = 0.5$, r = 100 no estado s6, r = 0 nos demais estados

Estado Inicial

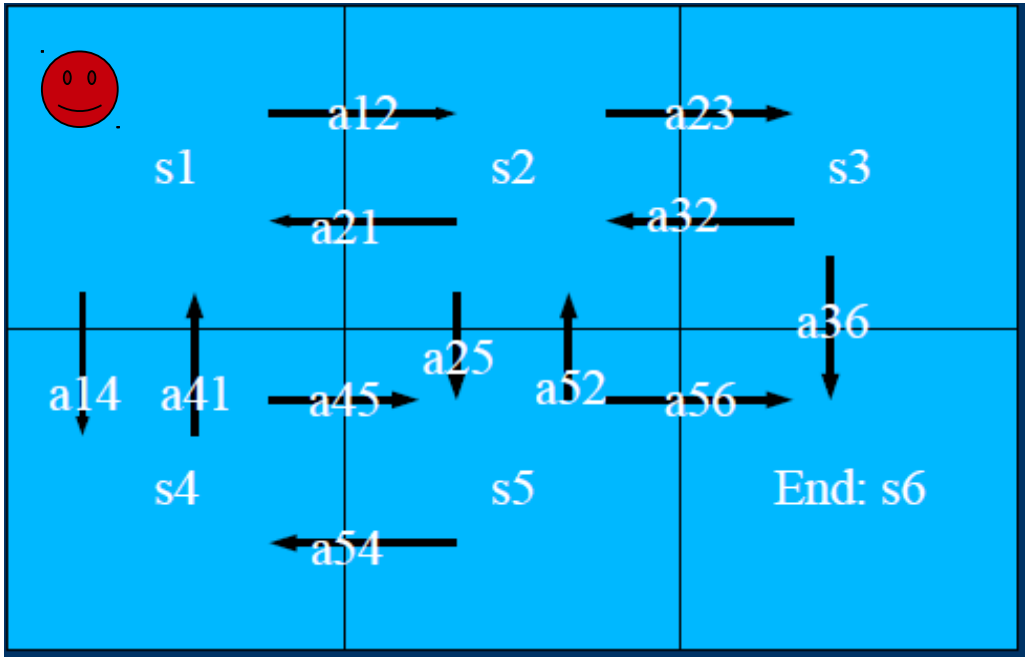
Q()	Recompensa
s1,a12	0
s1,a14	0
s2,a21	0
s2,a23	0
s2,a25	0
s3,a32	0
s3,a36	0
s4,a41	0
s4,a45	0
s5,a54	0
s5,a52	0
s5,a56	0



Apresentação do Algoritmo

Q()	Recompensa
s1,a12	0
s1,a14	0
s2,a21	0
s2,a23	0
s2,a25	0
s3,a32	0
s3,a36	0
s4,a41	0
s4,a45	0
s5,a54	0
s5,a52	0
s5,a56	0

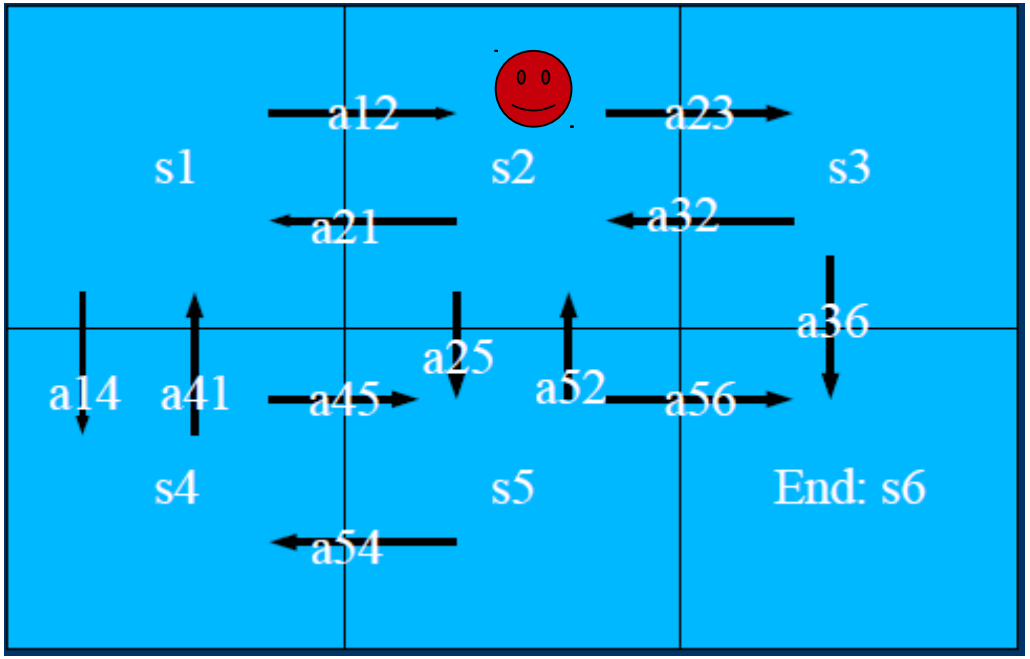
Estado atual: S1
Ações disponíveis: a12, a14
Escolha: a12



Atualiza Q(s1,a12)

Q()	Recompensa
s1,a12	0
s1,a14	0
s2,a21	0
s2,a23	0
s2,a25	0
s3,a32	0
s3,a36	0
s4,a41	0
s4,a45	0
s5,a54	0
s5,a52	0
s5,a56	0

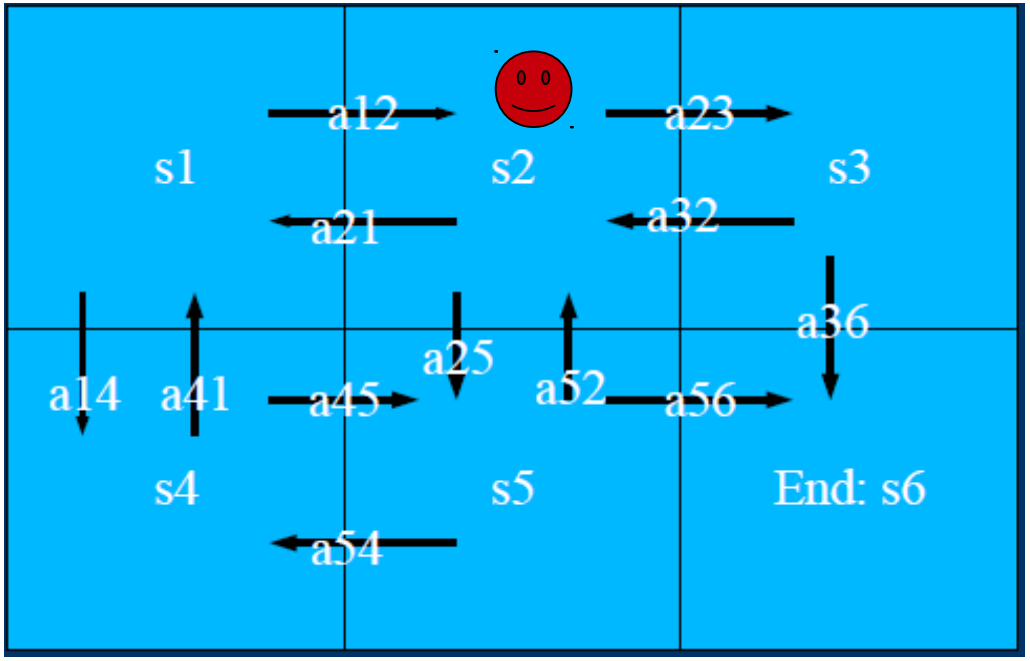
Estado atual: S2
Ações disponíveis: a21, a25, a23
Atualiza Q(s1, a12):
Q(s1, a12) = r + 0.5 * max(Q(s2,a21), Q(s2,a25), Q(s2,a23))
= 0



Próximo passo

Q()	Recompensa
s1,a12	0
s1,a14	0
s2,a21	0
s2,a23	0
s2,a25	0
s3,a32	0
s3,a36	0
s4,a41	0
s4,a45	0
s5,a54	0
s5,a52	0
s5,a56	0

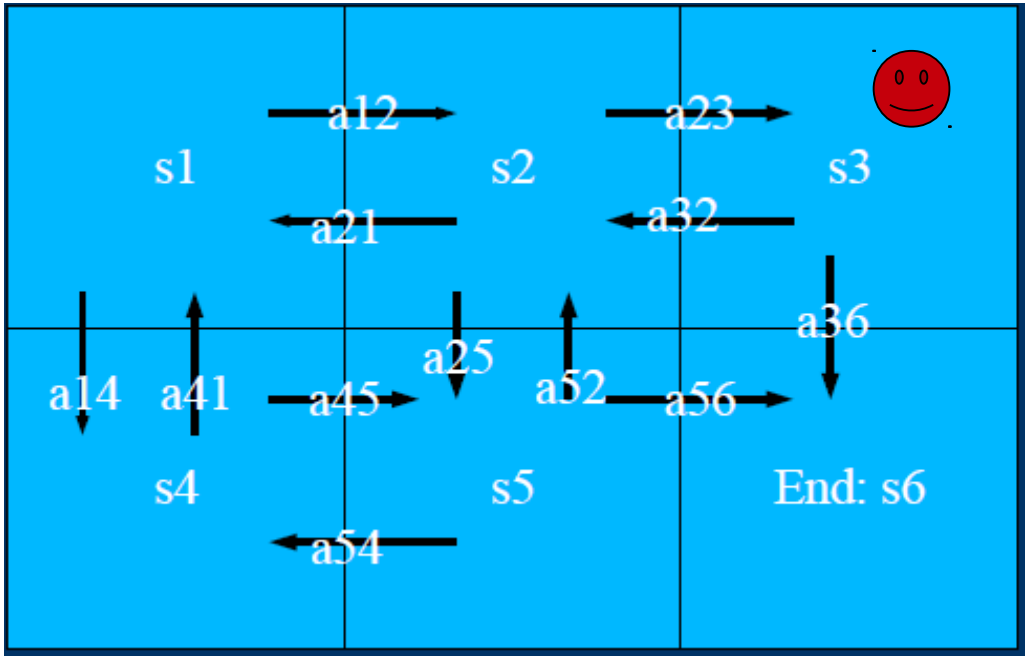
Estado atual: S2
Ações disponíveis: a21, a23,a25
Escolha: a23



Atualiza Q(s2,a23)

Q()	Recompensa
s1,a12	0
s1,a14	0
s2,a21	0
s2,a23	0
s2,a25	0
s3,a32	0
s3,a36	0
s4,a41	0
s4,a45	0
s5,a54	0
s5,a52	0
s5,a56	0

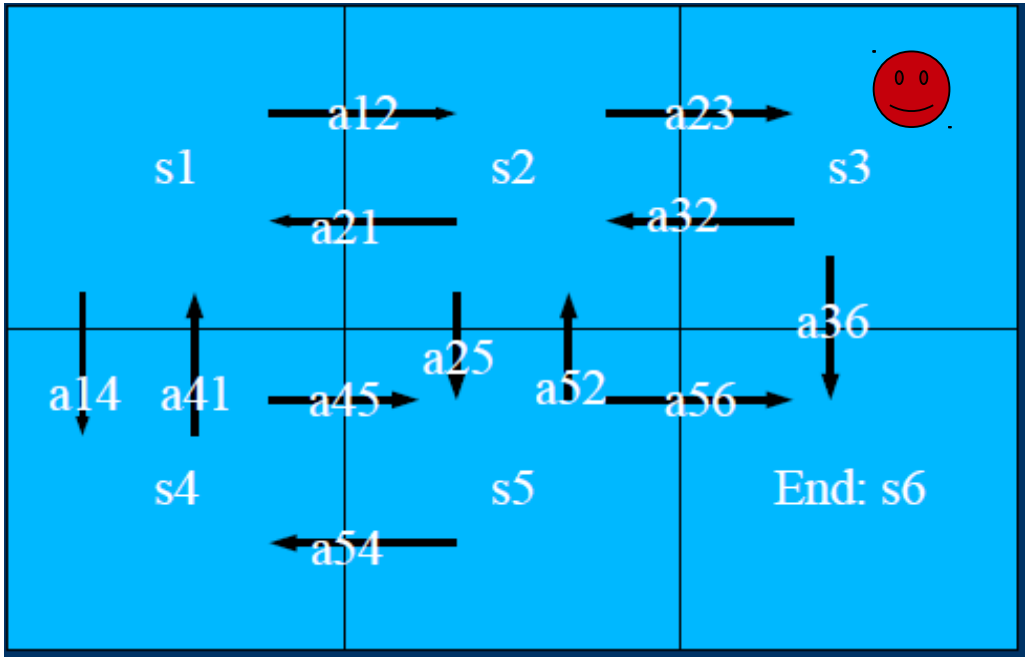
Estado atual: S3
Ações disponíveis: a32, a36
Atualiza Q(s2, a23):
Q(s2, a23) = r + 0.5 * max(Q(s3,a32), Q(s3,a36))
= 0



Próximo passo

Q()	Recompensa
s1,a12	0
s1,a14	0
s2,a21	0
s2,a23	0
s2,a25	0
s3,a32	0
s3,a36	0
s4,a41	0
s4,a45	0
s5,a54	0
s5,a52	0
s5,a56	0

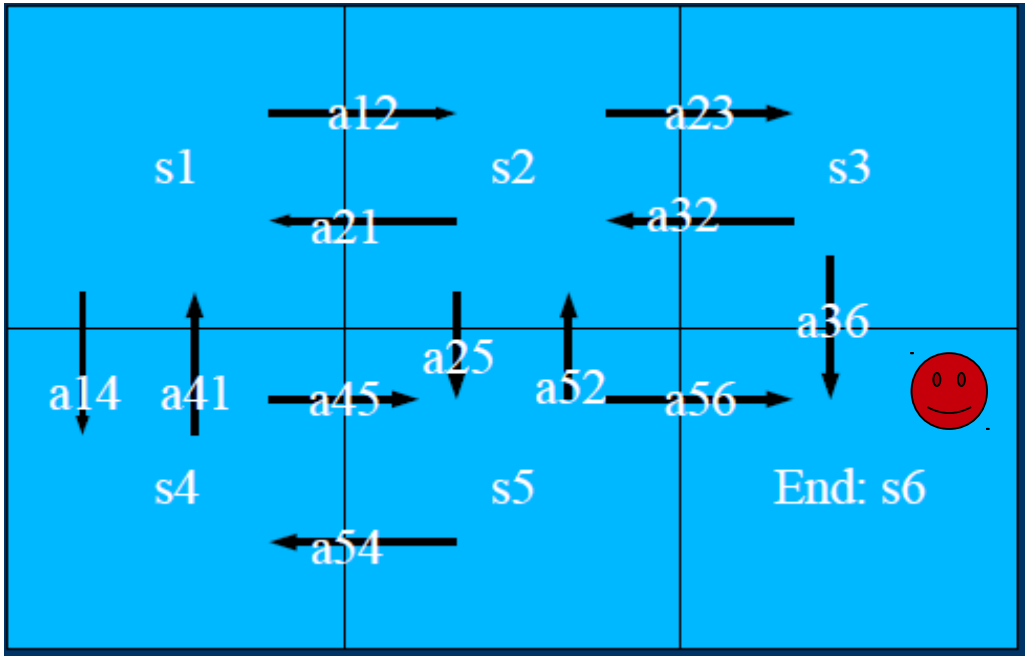
Estado atual: S3
Ações disponíveis: a32, a36
Escolha: a36



Atualiza Q(s3,a36)

Q()	Recompensa
s1,a12	0
s1,a14	0
s2,a21	0
s2,a23	0
s2,a25	0
s3,a32	0
s3,a36	100
s4,a41	0
s4,a45	0
s5,a54	0
s5,a52	0
s5,a56	0

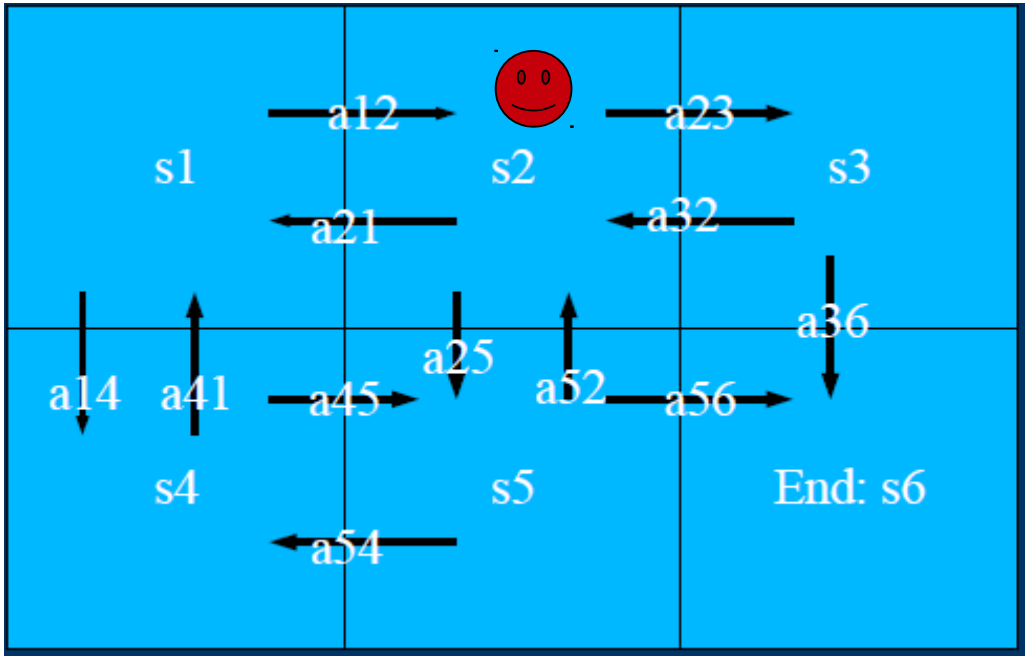
Estado atual: S6
Estado terminal
Atualiza Q(s3, a36):
Q(s3, a36) =100



Reinicia...

Q()	Recompensa
s1,a12	0
s1,a14	0
s2,a21	0
s2,a23	0
s2,a25	0
s3,a32	0
s3,a36	100
s4,a41	0
s4,a45	0
s5,a54	0
s5,a52	0
s5,a56	0

Estado atual: S2
Ações disponíveis: a21, a23,a25
Escolha: a23



Atualiza Q(s2,a23)

Q()	Recompensa
s1,a12	0
s1,a14	0
s2,a21	0
s2,a23	50
s2,a25	0
s3,a32	0
s3,a36	100
s4,a41	0
s4,a45	0
s5,a54	0
s5,a52	0
s5,a56	0

Estado atual: S3
Ações disponíveis: a32, a36
Atualiza Q(s2, a23):
Q(s2, a23) = r + 0.5 * max(Q(s3,a32), Q(s3,a36))
Q(s2, a23) = 0+0.5*100=50

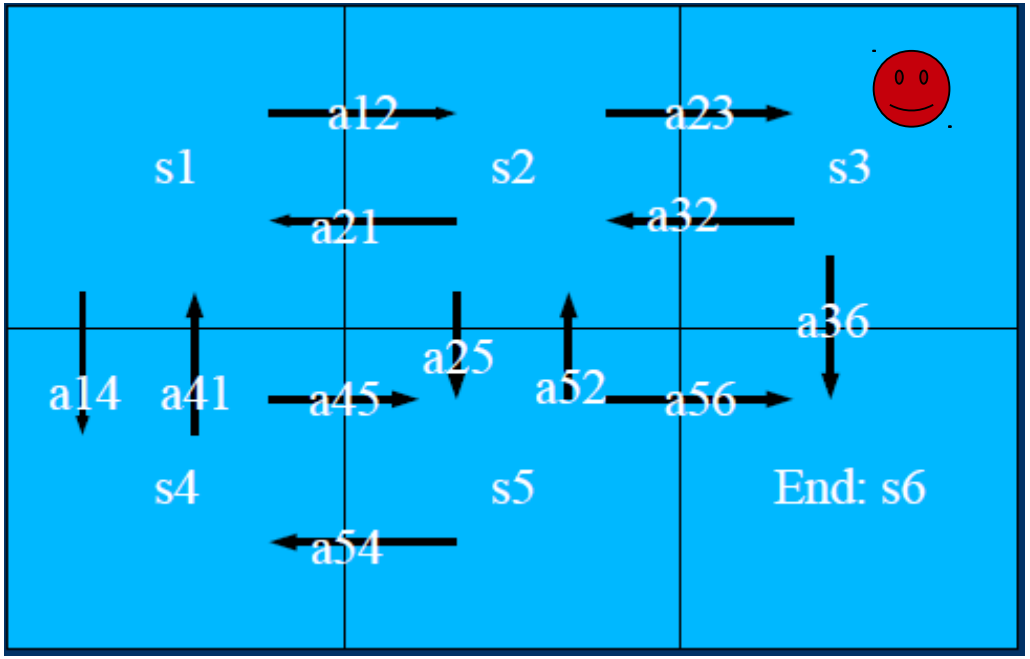
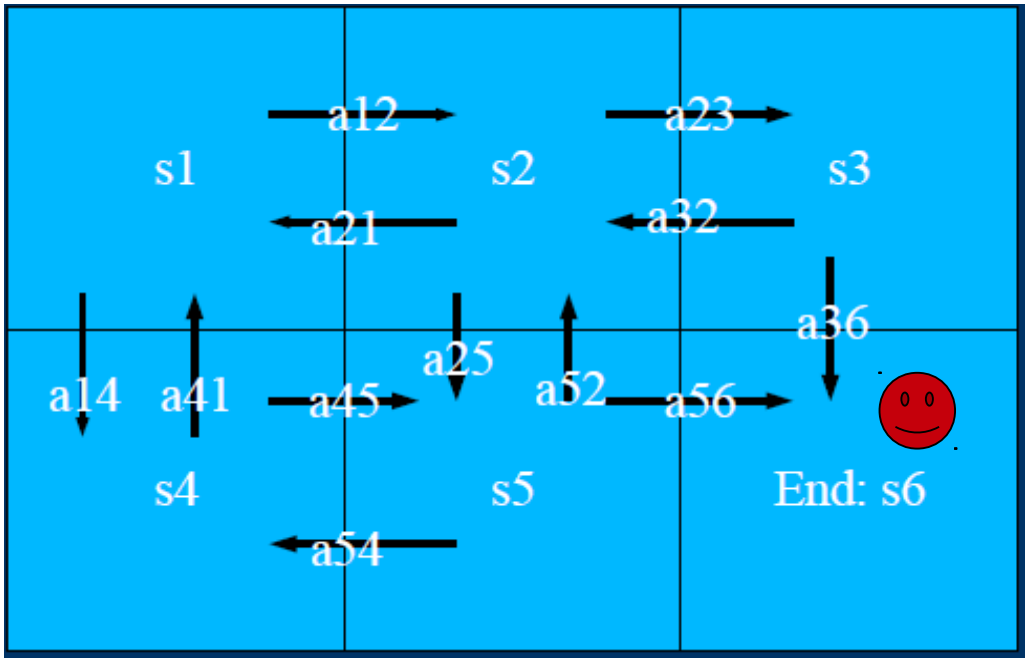


Tabela Q após aprendizagens...

Q()	Recompensa
s1,a12	25
s1,a14	25
s2,a21	12.5
s2,a23	50
s2,a25	50
s3,a32	25
s3,a36	100
s4,a41	12.5
s4,a45	50
s5,a54	25
s5,a52	25
s5,a56	100



Melhorias desejadas no trabalho

$$Q(s,a)=r(s,a)+\gamma max_{a'}(Q(s',a'))$$

Tem-se:

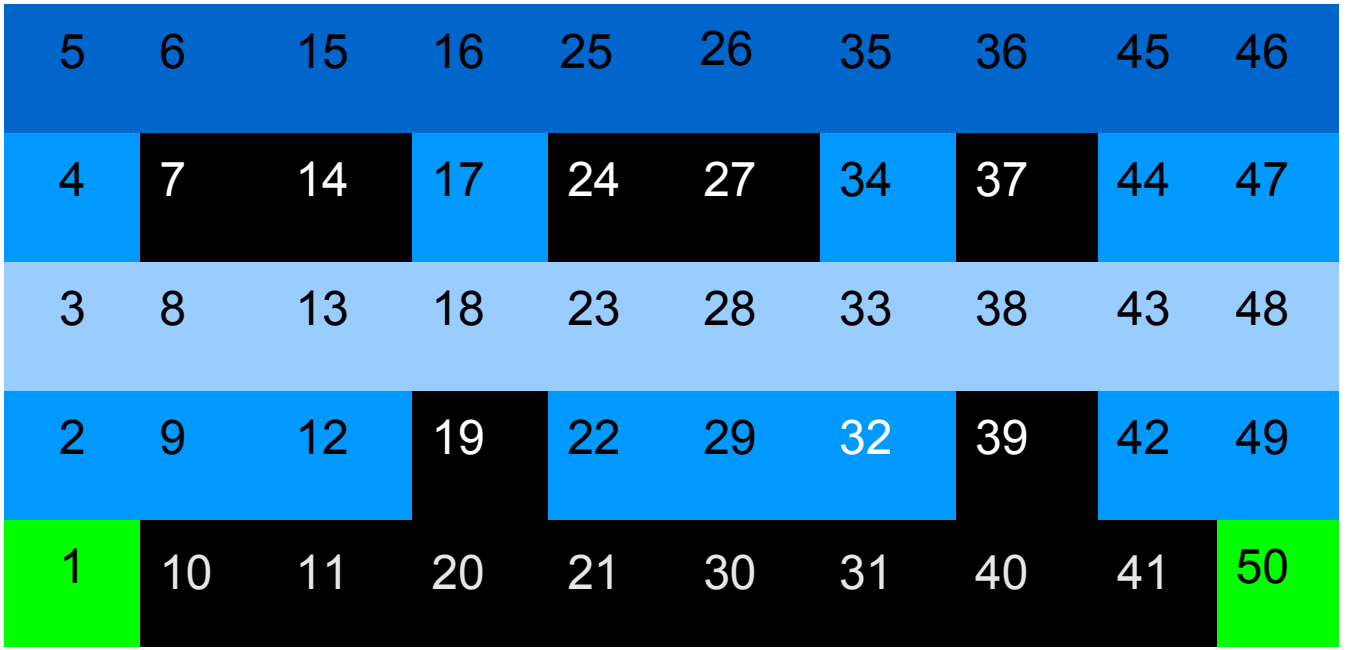
- gama é a taxa de propagação
- max deve considerar um valor randômico para evitar máximos locais:
    - Uma boa escolha, por exemplo, é escolher, em 70% dos casos, a ação que retorne o valor máximo e nos outros 30% faz-se escolhas aleatórias. (Os valores 70 e 30% devem ser testados.)

Tarefa

1. Realizar a modelagem do problema
2. Especificar estruturas de dados
3. Implementar o algoritmo Qlearning com interface gráfica para acompanhamento da execução
4. Testar e realizar correções necessárias
5. Documentar resultados conforme template a ser fornecido.
6. Apresentar os resultados da implementação em aula para a turma.

Mapa

O mapa a ser utilizado é o seguinte:



Estado inicial: 1
Estado final: 50
Recompensas:
R=100 para o estado 50
R=-1 para os estados azuis
R=-100 para estados pretos

Os estados pretos representam locais intransponíveis

Objetivo: chegar ao estado 50
Encontrar a política ótima
Ações: ir para norte, ir para sul, ir para leste, ir para oeste

Cronograma do Desenvolvimento

21-05: início do projeto

28-05: aula destinada ao desenvolvimento

04-06: aula destinada ao desenvolvimento

11-06: apresentação dos trabalhos

A implementação individualmente ou em duplas.