

Insper

Inteligência Artificial e Robótica

Design Review

Insper
Fabrício Barth

Objetivos

Ao final da disciplina o estudante será capaz de:

1. Descrever os conceitos, técnicas e métodos para o desenvolvimento de **Agentes Autônomos**.
2. Identificar **quais tipos de problemas podem ser resolvidos** através do uso de Agentes Autônomos.
3. Criar soluções para alguns **problemas clássicos** desta área.
4. Especificar, desenvolver e testar projetos que façam uso de Agentes Autônomos para resolver **problemas complexos**.
5. Planejar e executar um trabalho em equipe, fornecendo e assimilando devolutivas.

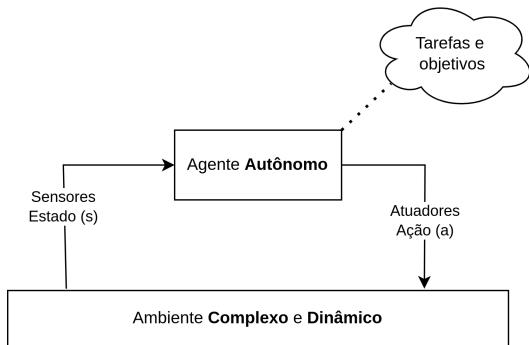
Como?

Primeira semana

Através de uma dinâmica de grupo, envolvendo **exemplos e contra-exemplos de aplicações de IA**, os estudantes irão:

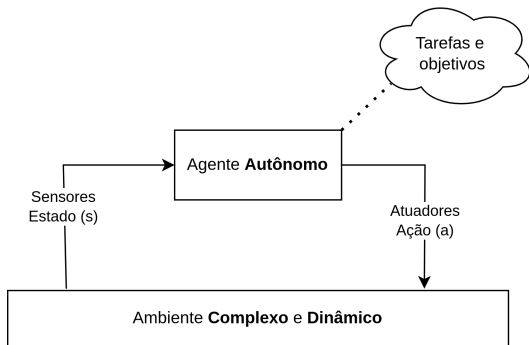
1. Criar uma definição para *strong AI* (*general AI*).
2. Enumerar diferenças entre *strong AI* e *weak AI*.
3. Criar uma definição para **agente autônomo**.
4. Listar capacidades que um agente autônomo deve ter (ementa desta disciplina e de disciplinas futuras).
5. Listar característica de ambientes.

Output da primeira semana



1. **Problem-solving**
2. Knowledge and reasoning
3. Learning
 - 3.1 Supervised Learning and Unsupervised Learning
 - 3.2 **Reinforcement Learning**
4. Communicating, perceiving, and acting
 - 4.1 Natural Language Processing
 - 4.2 Computer Vision
 - 4.3 **Robotics**

Output da primeira semana



Ambientes:

1. *Single Agent*, determinístico, custo uniforme
2. *Single Agent*, determinístico, custo não uniforme
3. *Single Agent*, não-determinístico
4. *Multi Agent Competitivo*, determinístico

Fora do escopo: *multi agent* colaborativo.

Ementa da disciplina

1. Definições de Agente Autônomo e resolução de problemas.
2. Estratégias de busca: algoritmos de busca cega e algoritmos informados.
3. Heurísticas.
4. Implementação de agentes autônomos utilizando estratégias de busca.
5. Programação por restrições (CSP).
6. Ambientes competitivos e teoria de jogos.
7. Algoritmo Min-Max e função de utilidade.
8. Implementação de agentes autônomos para ambientes competitivos.
9. Aprendizagem por Reforço.
10. Implementação de agentes autônomos usando aprendizagem por reforço.
11. Algoritmo Q-Learning.
12. Implementações de agentes autônomos usando o projeto Gym.
13. Implementação de um agente robótico.

Primeiro bloco: busca, busca informada e busca competitiva

Questões e Exemplos de Problemas

- ▶ O que é relevante representar nos estados do mundo? Como os estados são estruturados (estrutura de dados) e qual o significado dela (dos campos)?
- ▶ Quais as operações sobre os estados?
- ▶ Qual a estimativa do tamanho do espaço de busca?
- ▶ Qual é a estimativa da árvore de busca?
- ▶ É necessário utilizar um algoritmo informado? Se sim, qual é a heurística? Existe heurística admissível?
- ▶ Aspirador de pó
- ▶ O homem, o lobo, o carneiro e o cesto de alface
- ▶ Banda U2 - problema de custo não uniforme
- ▶ Cavalo e tabuleiro de xadrez
- ▶ Problema das N rainhas

Implementação (1/2)

ProblemSpecificationExample.py M X

src > ProblemSpecificationExample.py > ...

```
1  from aicode.search.SearchAlgorithms import BuscaProfundidadeIterativa
2  from aicode.search.Graph import State
3
4  class ProblemSpecifcation(State):
5
6      def __init__(self, op):
7          self.operator = op
8          #TODO
9
10     def sucessors(self):
11         sucessors = []
12         #TODO
13         return sucessors
```

Implementação (2/2)

```
15 def is_goal(self):
16     pass
17
18 def description(self):
19     return "Descrição do problema"
20
21 def cost(self):
22     return 1
23
24 def print(self):
25     #
26     # Usado para imprimir a solução encontrada.
27     # O PATH do estado inicial até o final.
28     return str(self.operator)
29
30 > def env(self): ...
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48 def main():
49     print('Busca em profundidade iterativa')
50     state = ProblemSpecification('')
51     algorithm = BuscaProfundidadeIterativa()
52     result = algorithm.search(state)
53     if result != None:
54         print('Achou!')
```

Conceitos explorados e a relação com outras disciplinas

- ▶ Algoritmos de Busca: Busca em Largura e Profundidade → **Técnicas de Programação**
- ▶ Recursão → **Técnicas de Programação**
- ▶ Busca em Profundidade Iterativa, Busca de Custo Uniforme, Busca Gananciosa, Busca A*
- ▶ Grafos e Árvores → **Algoritmos e Estrutura de Dados**
- ▶ Heurística
- ▶ Algoritmos de subida da montanha e outros relacionados com CSP
- ▶ Algoritmo Min-Max

Segundo bloco: Aprendizagem por Reforço

Aprendizagem por Reforço (RL)

- ▶ Para tratar problemas não determinísticos
- ▶ Heurística e Função de Utilidade que são aprendidas pelo próprio agente
- ▶ Agente Autônomo, Ambiente e **Reforço**
- ▶ Algoritmo *Q-Learning*
- ▶ Exemplo de programação dinâmica

Implementação

```
import gymnasium as gym
from QLearning import QLearning

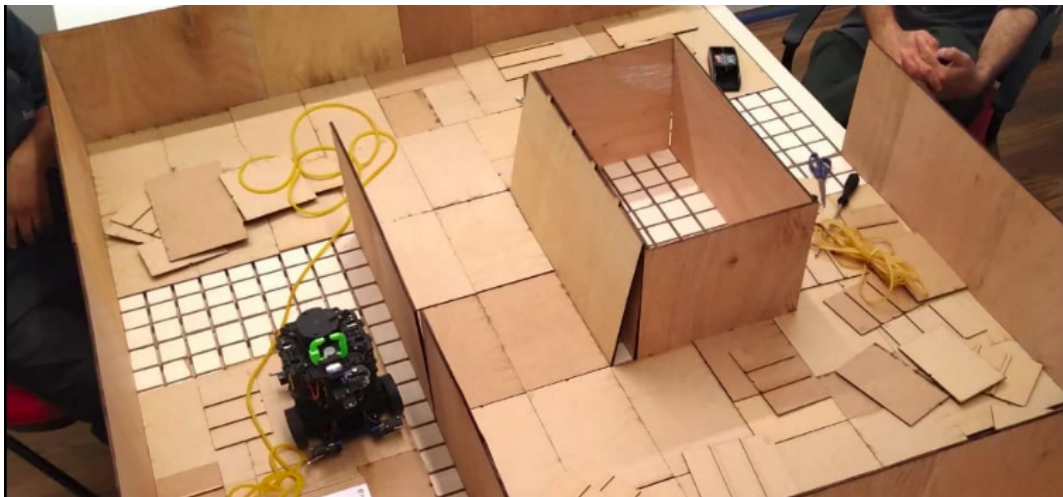
env = gym.make("Taxi-v3", render_mode='ansi').env
env = gym.make('FrozenLake-v1', render_mode='ansi').env
env = gym.make("Blackjack-v1", render_mode='ansi')

qlearn = QLearning(env, alpha=0.1, gamma=0.6, epsilon=0.7, epsilon_min=0.0)
q_table = qlearn.train('data/q-table-taxi-driver.csv', 'results/actions_taxi')

from kaggle_environments import make
env = make("tictactoe", debug=True)
```

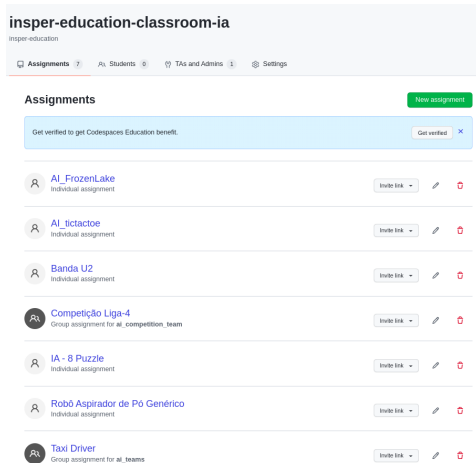
Terceiro bloco: Robótica

Implementação de um agente robótico



Avaliação

- ▶ Todos os projetos utilizam o Github Classroom.
- ▶ Alguns projetos fazem uso de testes automatizados.
- ▶ Alguns projetos são individuais, outros em grupo.



The screenshot displays the 'insper-education-classroom-ia' interface. At the top, there's a navigation bar with 'insper-education-classroom-ia' and 'insper-education' below it. The main menu includes 'Assignments' (active), 'Students', 'TAs and Admins', and 'Settings'. A green 'New assignment' button is in the top right. Below the menu, a blue banner reads 'Get verified to get Codespaces Education benefit.' with a 'Get verified' link. The main content area lists seven assignments:

Assignment Name	Type	Invite Link	Edit	Delete
AI_FrozenLake	Individual assignment	Invite link	✎	🗑️
AI_tictactoe	Individual assignment	Invite link	✎	🗑️
Banda U2	Individual assignment	Invite link	✎	🗑️
Competição Liga-4	Group assignment for ai_competition_team	Invite link	✎	🗑️
IA - 8 Puzzle	Individual assignment	Invite link	✎	🗑️
Robô Aspirador de Pó Genérico	Individual assignment	Invite link	✎	🗑️
Taxi Driver	Group assignment for ai_teams	Invite link	✎	🗑️