



*Escola Politécnica da USP - Depto. de Enga. Mecatrônica*

# PMR-3510 Inteligência Artificial

## Aula 1 - Introdução

*Prof. José Reinaldo Silva*

*[reinaldo@usp.br](mailto:reinaldo@usp.br)*





## *PMR 3510 - Inteligencia Artificial*

### Módulo 1

- 1.1 Introdução, Agentes Inteligentes, A sociedade da Mente (Minsky), Aplicações da IA em automação (Aula 1);
- 1.2 Resolução de Problemas, Métodos clássicos de busca (Aula 2 e 3);
- 1.3 Busca Informada (Aulas 4 e 5)

### Módulo 2

- 2.1 Sistemas baseados em conhecimento, representação de conhecimento (Aula 6);
- 2.2 Introdução à Lógica de 1a. ordem (Aula 7 e 8);
- 1.3 Métodos de inferência, exercícios (Aulas 9 e 10);



## *PMR 3510 - Inteligencia Artificial*

### Módulo 3

3.1 Planejamento inteligente (Aula 11 e 12);

3.2 Problemas e casos práticos de planejamento, STRIPS (Aula 14 e 15);

3.3 Ações e planejamento (Aulas 16)

### Módulo 4: Aplicações

#### Avaliação:

Lista de exercícios(25%); Trabalho em grupo (2)(25%); prova no final do curso (50%)

Acompanhamento do curso e aprendizado paralelo de programação Prolog via e-disciplinas e SWI-online

Livro texto: Artificial Intelligence: a new synthesis, Nils Nilsson (2009); Artificial Intelligence: a modern approach, Stuart Russel, Peter Norvig (3rd. ed.) (2009) – site AIMA, UC Berkeley.

Apoio: Programming in Prolog, W. F. Clocksin, C. S. Mellish (5a. ed.) (2003).



## *PMR 3510 - Inteligencia Artificial*

Livro texto: Artificial Intelligence: a new synthesis, Nils Nilsson (2009);  
Artificial Intelligence: a modern approach, Stuart Russel, Peter Norvig (3rd. ed.) (2009) - site AIMA, UC Berkeley.

Apoio: Programming in Prolog, W. F. Clocksin, C. S. Mellish (5a. ed.) (2003).

### **Apoio Web:**

Site e-disciplinas (PMR 3510): o acompanhamento da disciplina, material de apoio, exercícios, trabalhos, tópicos de programação, serão repassados e recebidos somente através deste site.

Programação em Prolog: [www.learnprolognow.org](http://www.learnprolognow.org)

Sistema para programação: [swish.swi-prolog.org](http://swish.swi-prolog.org) (SWI Prolog)

# Learn Prolog Now!

by Patrick Blackburn, Johan Bos, and Kristina Striegnitz

> LPN! Home

Free Online Version

Paperback English

Paperback Français

Teaching Prolog

Learn Prolog Now! is an introductory course to programming in [Prolog](#). The [online version](#) has been available since 2001, and now there is also a thoroughly revised version available in [book form](#).

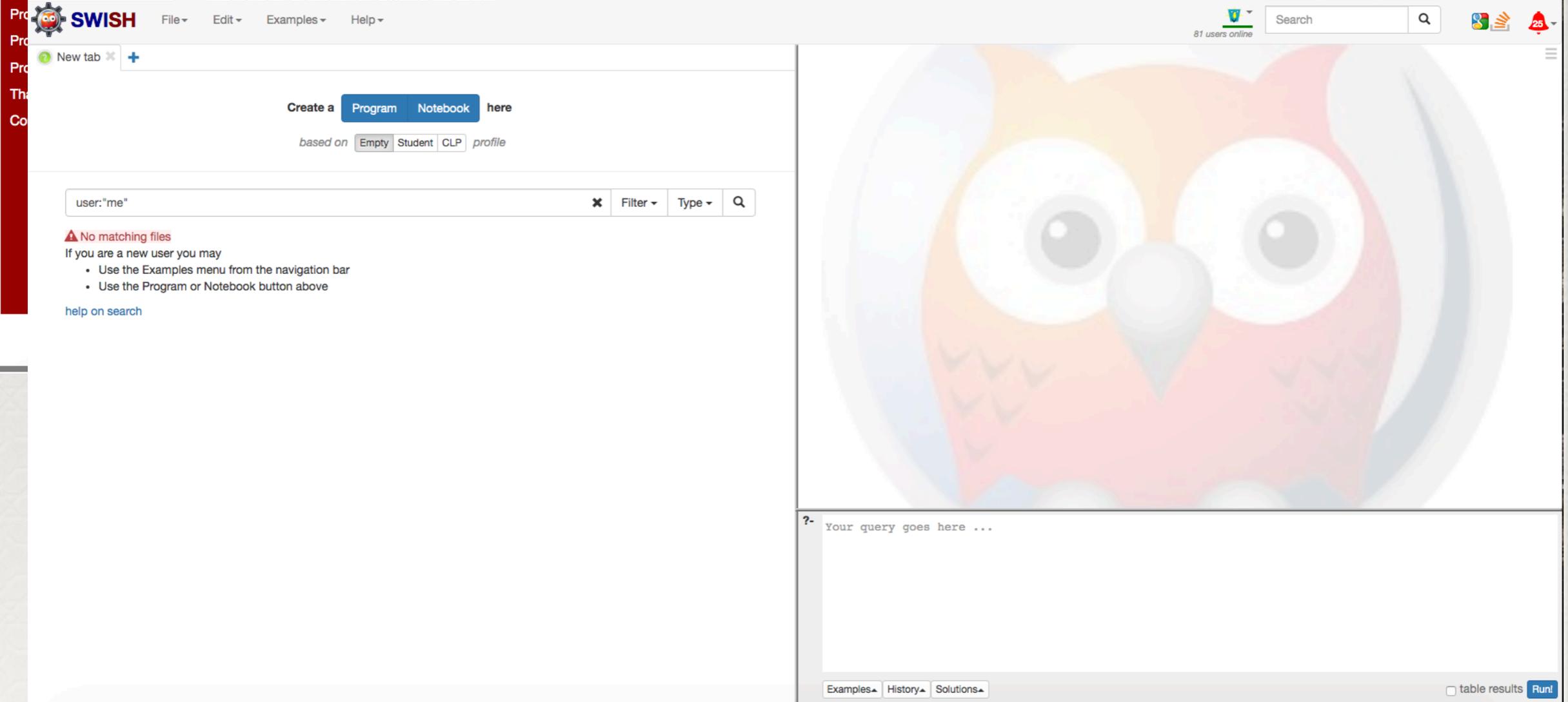
We wanted to do two things with this course. First, we wanted to provide a text that was relatively self contained, a text that would permit someone with little or no knowledge of computing to pick up the basics of Prolog with the minimum of fuss. We also wanted the text to be clear enough to make it useful for self study. We believe that if you read the text, and do the associated exercises, you will gain a useful partial entry to the world of Prolog.

But only a partial entry, and this brings us to our second point. We want to emphasize the practical aspects of Prolog. Prolog is something you do. You can't learn a programming language simply by reading about it, and if you really want to get the most out of this course, we strongly advise you to get hold of a Prolog interpreter (you'll find pointers to some nice ones on this website) and work through all the Practical Sessions that we provide. And of course, don't stop with what we provide. The more you program, the better you'll get....

We hope you enjoy the course. And whether you're using this book to teach yourself Prolog, or you're using it as the basis for teaching others, we would like to hear from you. Please [send us](#) any comments/corrections you have so that we can take them into account in later versions.

## Tweets by @learnprolognow

 **Learn Prolog Now!**  
@learnprolognow  
Interesting new interactive  
(you can run and modify ex



SWISH File Edit Examples Help

81 users online Search

Create a **Program** **Notebook** here

based on **Empty** **Student** **CLP** profile

user:me Filter Type

**No matching files**

If you are a new user you may

- Use the Examples menu from the navigation bar
- Use the Program or Notebook button above

[help on search](#)

?- Your query goes here ...

Examples History Solutions  table results **Run!**



## *Introdução*

### *Historia da Inteligência Artificial*

Desde o século XIII (com o trabalho de Ramon Lull) existe uma grande preocupação para entender melhor a "inteligência", e o diferencial humano. Em várias ocasiões surgiram propostas para entender, controlar e até reproduzir o processo pensante (incluindo a teoria da racionalidade de Herbert Simon e Allen Newel), quase sempre embebido de motivação filosófica, psicológica e/ou religiosa. Vamos nos ater às tentativas mais recentes e científicas...



## *Introdução*

*Existem basicamente dois “modelos de inteligência” que serviram de base para as tentativas de reprodução artificial de um processo racional:*

- ✦ **uma máquina capaz de emular um jogador de xadrez**
- ✦ **uma máquina capaz de fazer cálculos complexos ou demonstrações de teorema**



## Introdução

- ✦ a máquina capaz de emular um jogador de xadrez



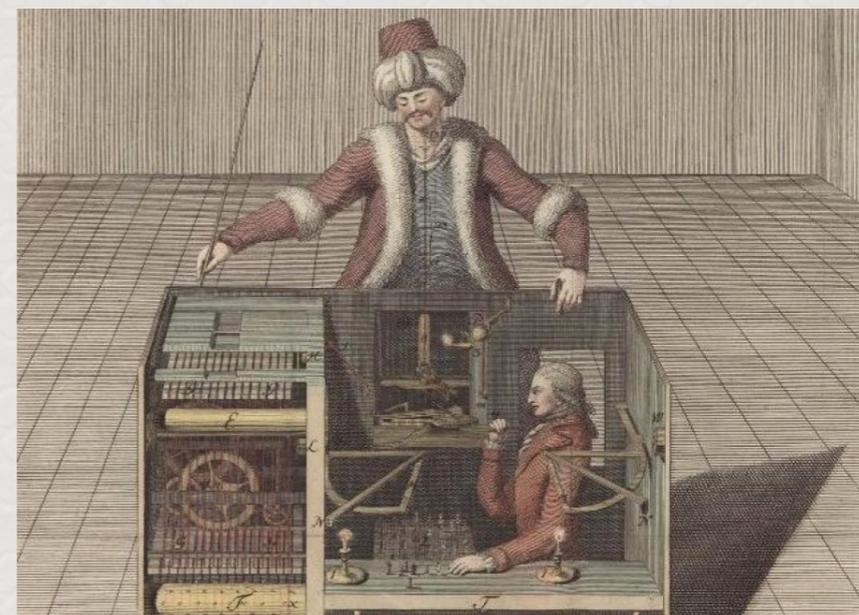
A “máquina inteligente” (autômato) conhecida como “o Turco”, foi criada na Austria-Hungria em 1770 por Wolfgang Von Kepelen, para agradar a duquesa de Habsburgo, Maria Theresa. O Turco ficou famoso por vencer vários jogador habilidosos no período 1770-1789.



## Introdução

- a máquina capaz de emular um jogador de xadrez

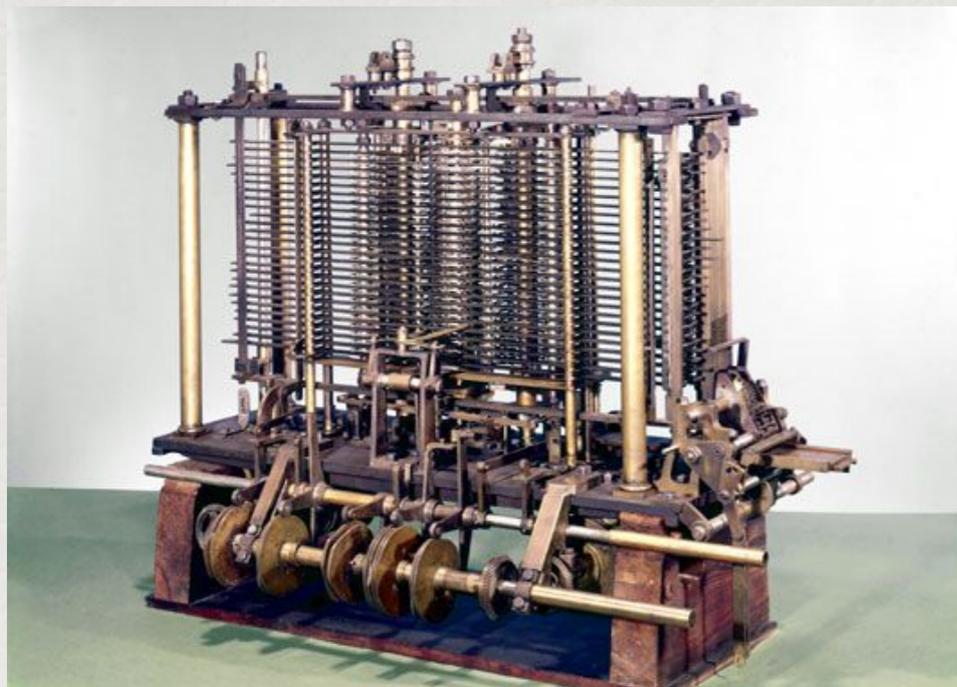
Na verdade o Turco não passava de um truque e tinha de fato um anão - exímio jogador de xadrez - dentro da mesa, que fazia os movimentos com algo parecido com um "mouse 3D".





## Introdução

- ✦ a máquina capaz de fazer cálculos complexos ou demonstrações de teorema



Charles Babbage foi um matemático inglês nascido justamente em 1791 que projeto uma máquina para fazer cálculos astronômicos, que mais tarde se tornou a "Máquina Diferencial 2", projetada em 1849, após mudanças inseridas por Ada Augusta King (nascida Ada Byron), condessa de Lovelace. Charles Babbage e Ada King são considerados os "pais do computador moderno".



## Introdução



Enigma

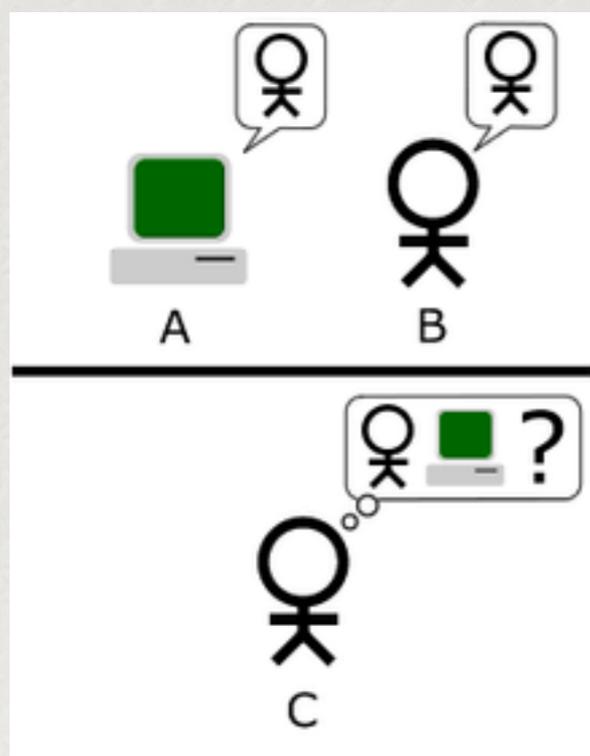


O primeiro modelo matemático do processo de computação foi proposto por Alan Turing em 1937. Fiel à tendência mostrada aqui, o próprio Turing propôs uma máquina chamada ACE (Automatic Computing Engine) – homenagem a Charles Babbage – em 1945 usando uma linguagem de programação chamada Plankalkul. Esta seria a base para a sua máquina (1941) usada para decifrar o código gerado pela máquina alemã Enigma, o que contribuiu muito para que os aliados vencessem a segunda guerra mundial.



## Introdução

linguagem natural  
representação de conhecimento  
raciocínio automático  
aprendizado de máquina



Em 1950 Turing propôs um teste que definiria a capacidade e o resultado esperado do que veio a se chamar Inteligência Artificial. Um sistema inteligente seria aquele que conseguisse passar no teste de Turing, que consistia basicamente no seguinte: um entrevistador humano dirigiria perguntas via terminal que poderiam ser respondidas por um humano ou por outro computador. Se o entrevistador não conseguisse distinguir entre as respostas dadas pelo computador ou pelo agente humano então este sistema computacional seria definitivamente "inteligente".



## *Introdução*



O tema do processamento computacional de processos racionais continuou na pauta de vários pesquisadores e engenheiros até que em 1956, John McCarthy propôs um evento com os principais interessados neste tema no Dartmouth College, financiado pela Rockefeller Foundation. Os organizadores foram, além de McCarthy, Marvin Minsky (Harvard), Nathaniel Rochester (IBM) e Claude Shannon (Bell Labs). O termo "Inteligência Artificial" foi usado pela primeira vez neste evento com o significado atual.



## *Introdução*

*Desta curta e resumida digressão histórica podemos concluir que a moderna inteligência artificial nasceu baseada em dois pilares:*

- ✦ a reprodução de processos racionais, como a capacidade de jogar xadrez, de deduzir, planejar e aprender e “tomar decisões”;
- ✦ o uso de autômatos, ou máquinas, capazes de abrigar os processos acima, por exemplo, um computador.



## *Introdução*

*Esta disciplina é justamente sobre processos racionais baseados em regras e lógica de primeira ordem, que podem ser aplicadas em jogos simples (oito rainhas, torre de Hanoi, jogo da velha), deduções em modus ponens, ou em planejamento automático, isto é, traduzida em uma inteligência de máquina. Aprendizado e tomada de decisão serão vistos na disciplina PMR 3508, ministrada pelo Prof. Fabio Cozman. O nosso foco será portanto...*

- ✦ a reprodução de processos racionais, como a capacidade de jogar xadrez, de deduzir, planejar ações que servem de base para a automação de processos;
- ✦ como usar autômatos, ou máquinas capazes de abrigar os processos acima, no caso, um computador.



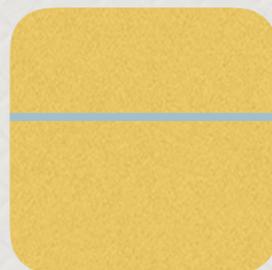
## Introdução

### *A evolução desde dados estáticos até os agentes inteligentes*

Portanto temos como objetivo representar conhecimento e processos racionais de forma computável. Assim, temos como ponto de partida a forma mais simples de conhecimento: dados – especialmente se tidos como confiáveis e verdadeiros. Mesmo a programação convencional tem dedicado muita atenção à representação de dados, que podem ser usados de forma estática ou interagindo com processos dinâmicos. A evolução básica nesta área é mostrada abaixo:



dados simples



módulos



objetos (frames)



agentes inteligentes

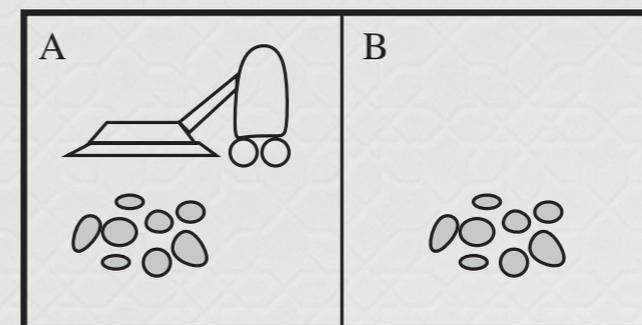
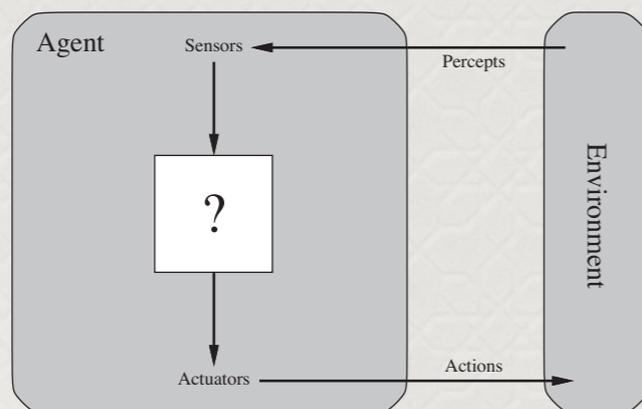


## Introdução



### *A evolução desde dados estáticos até os agentes inteligentes*

Um agente inteligente é capaz de perceber o que acontece no seu domínio de trabalho, ou environment, e definir que ação deve tomar ante as circunstâncias, atuando sobre este mesmo environment. A ação deve então ser inteligente e sua performance é medida pela capacidade de inferência do agente. Um exemplo simples seria um aspirador (inteligente) que detecta e localiza o pó a ser removido, traça um caminho (ótimo) até ele evitando obstáculos e efetivamente faz a remoção.



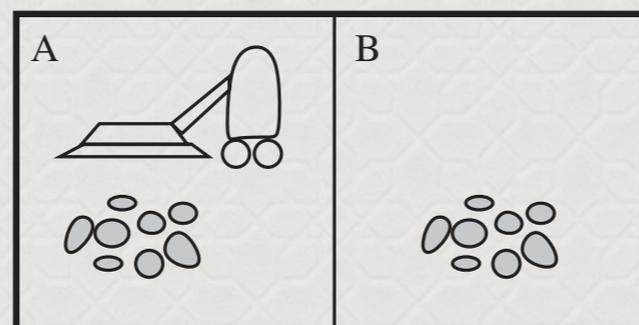


## Introdução



### *A evolução desde dados estáticos até os agentes inteligentes* *Agentes inteligentes autônomos*

Se existir uma relação simples entre percepção e ação (uma tabela de dupla entrada), teremos um agente de controle simples. Mas, se existir um processamento baseado em conhecimento entre percepção e ação o agente é dito inteligente. Este agente é dito autônomo se o seu comportamento é determinado exclusivamente pela sua própria relação com o ambiente e dados acumulados. O exemplo do aspirador pode ser usado para ir de um extremo a outro.



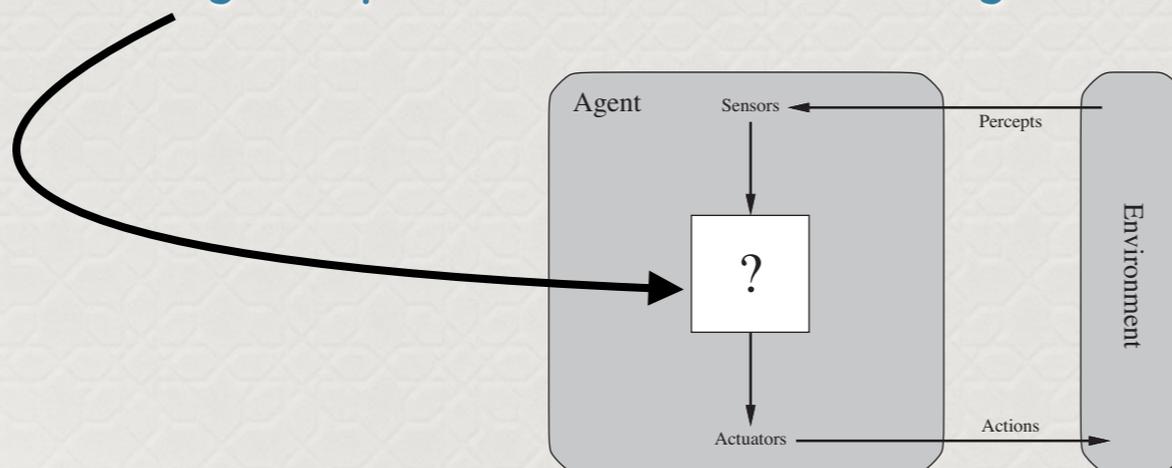


## Introdução

### *A evolução desde dados estáticos até os agentes inteligentes*

*Agentes inteligentes autônomos*

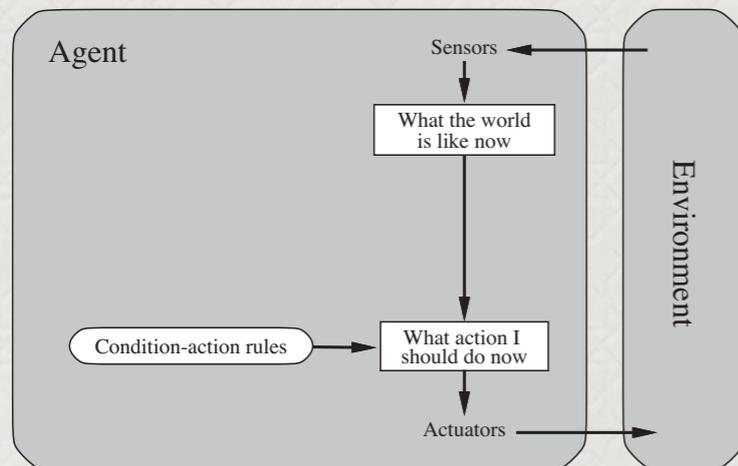
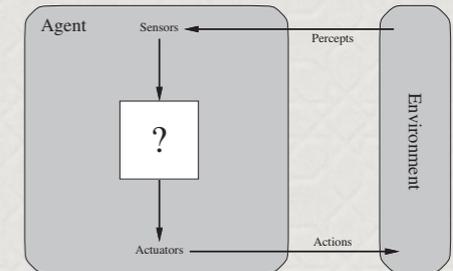
Até então definimos o agente inteligente pela sua relação com o ambiente que o cerca (environment). Note-se que os mecanismos de percepção e ação podem envolver hardware (sensores e atuadores). Mas a autonomia está ligada ao processamento (a interrogação na figura a seguir), que deve ser necessariamente executada por uma arquitetura computacional, que por sua vez tem os processos guiadas por um programa. Portanto, o nosso agente pode ser definido como: agente = arquitetura + programa.



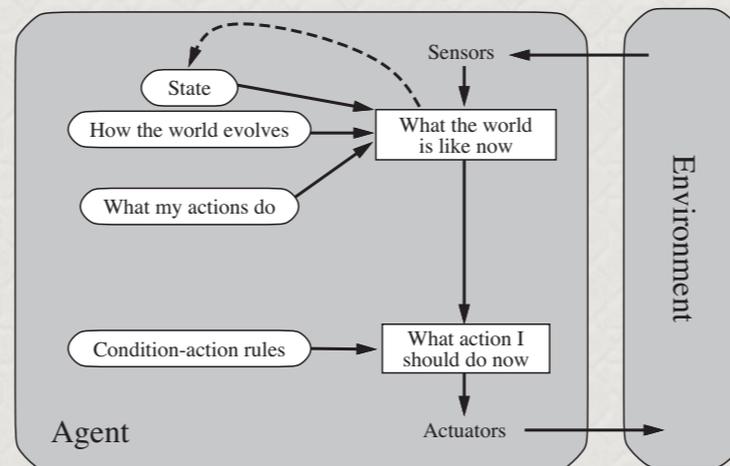


# Introdução

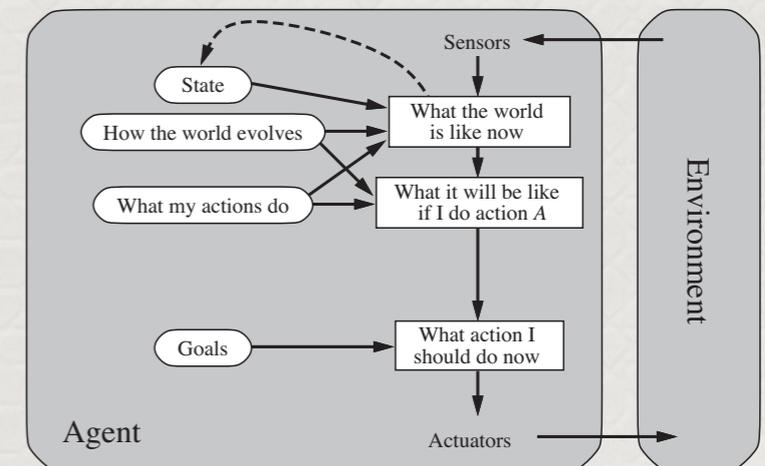
## From Data to Intelligent Agents *Agentes inteligentes autônomos*



Agente reflexo



Agente baseado em modelo



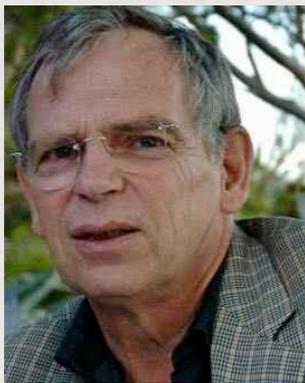
Agente baseado em objetivos

*Só mais um detalhe antes de encerrarmos a aula de hoje...*

*...seja qual for o agente e sua arquitetura precisamos ser capazes de mudar sua programação (as ações que este executa) sem precisar mexer no hardware. Como propôs Ada (Byron) King, precisamos de uma linguagem de comunicação com o agente inteligente.*

*Sem dúvida esta linguagem tem que ser capaz de expressar “conhecimento”, isto é, ações que não são puramente comandos (procedurais) de forma sucinta e fácil de interpretar. A base matemática pra isso é a Lógica, e, um sub-conjunto da lógica de primeira ordem pode ser transformada em linguagem interpretada para servir a este propósito.*

*A linguagem Prolog (de Programação Lógica) foi criada para servir ao propósito de expressar de forma declarativa (e não procedural) ações cognitivas, dentro de um sub-conjunto da lógica de primeira ordem. Foi criada por Alain Colmerauer e Robert Kowalski e lançada em 1972.*



*Alain Colmerauer (1941-2017)*



*Robert Kowalski (1945-2017)*

```
mother_child(trude, sally).
```

```
father_child(tom, sally).  
father_child(tom, erica).  
father_child(mike, tom).
```

```
sibling(X, Y) :- parent_child(Z,  
X), parent_child(Z, Y).
```

```
parent_child(X, Y) :- father_child(X,  
Y).  
parent_child(X, Y) :- mother_child(X,  
Y).
```

## *Aplicações práticas*

*“pig” robótico*

*Controle de água e reservatórios*

*Sistemas logísticos*

*Planejamento em companhias aéreas*

*Sistemas domóticos*

*...*



*Na aula que vem continuaremos a discussão  
agora enfocando a ação dos agentes  
inteligentes para resolver problemas.*

[Ativar edição](#)

## Administração

### » Administração do ambiente

[✖ Editar configurações](#)[✏ Ativar edição](#)[✖ Conclusão de curso](#)[» Usuários](#)[▾ Filtros](#)[» Relatórios](#)[✖ Configuração das Notas](#)[📊 Resultado da aprendizagem](#)[» Emblemas](#)[📁 Backup](#)[📁 Restaurar](#)[📁 Importar](#)[» Banco de questões](#)[🗑 Lixeira](#)[📄 Navegação ▾](#)

## Geral

Benvindos ao curso de Inteligência Artificial

Começaremos as aulas nesta próxima quarta-feira, 7 de agosto, e usaremos a sala MZ-02, que fica no piso superior, Bloco B (o bloco mais novo onde ficam os laboratórios). Esta sala fica logo defronte de quem sai da passarela que liga o Bloco A ao Bloco B e é a primeira sala à direita de quem entra neste corredor. A primeira sala à esquerda, antes de entrar no corredor é o D-Lab, ou Design Lab, o meu laboratório de pesquisa.

Faremos algumas pequenas mudanças com relação ao curso do ano passado, começando por enriquecer a bibliografia com um livro-texto adicional. Mas, o importante é que continuamos com o objetivo de estudar os conceitos da Inteligência Artificial e da inteligência de máquina, um tópico importante tanto do ponto de vista dos processos de automação como em todos os procesos que geralmente envolvem pessoas e máquinas.

Começaremos a discutir isso já nesta primeira aula.

[📧 Avisos](#)[📧 PMR-3510 Inteligência Artificial](#)[📄 Inteligência Artificial](#)

## Administração

### Administração do ambiente

Editar configurações

Ativar edição

Conclusão de curso

### Usuários

Usuários inscritos

Métodos de inscrição

Grupos

Permissões

Outros usuários

### Filtros

Relatórios

Configuração das Notas

Resultado da aprendizagem

Emblemas

Backup

Restaurar

Importar

Banco de questões

Lixeira

Navegação

## Participantes

Não foram aplicados filtros

Pesquise por palavra-chave ou selecione um filtro

Inscriver usuários

Número de participantes: 35

Nome **Todos** A B C D E F G H I J K L M N O P Q R S T U V W X Y Z  
Sobrenome **Todos** A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

1 2 »

Selecionar	Nome / Sobrenome	Endereço de email	Número USP	Papéis	Grupos	Último acesso ao curso	Estado
<input type="checkbox"/>	 Alexandre Zamora Zerbini Denigres	alexandre.denigres@usp.br	8583072	Estudante	T-PMR3510-2019250	Nunca	Ativo
<input type="checkbox"/>	 Beatriz Santin	beatriz.pinho@usp.br	8533851	Estudante	T-PMR3510-2019250	Nunca	Ativo
<input type="checkbox"/>	 Bianca Faria Silva	bianca.faria.silva@usp.br	8991599	Estudante	T-PMR3510-2019250	Nunca	Ativo
<input type="checkbox"/>	 Bruna Sayuri de Souza Suzuki	bruna.suzuki@usp.br	7987501	Estudante	T-PMR3510-2019250	7 horas 45 minutos	Ativo
<input type="checkbox"/>	 Daniel Tsutsumi	daniel.tsutsumi@usp.br	9349005	Estudante	T-PMR3510-2019250	Nunca	Ativo
<input type="checkbox"/>	 Danilo Polidoro	danilo.polidoro@usp.br	8582982	Estudante	T-PMR3510-2019250	Nunca	Ativo
<input type="checkbox"/>	 David Calil Spindola Pedro	david.pedro@usp.br	8989384	Estudante	T-PMR3510-2019250	Nunca	Ativo
<input type="checkbox"/>	 Diego Augusto Vieira Rodrigues	diego.augusto.rodrigues@usp.br	8989276	Estudante	T-PMR3510-2019250	Nunca	Ativo
<input type="checkbox"/>				Estudante	T-PMR3510-		Ativo



*Acessem o site da disciplina e vejam o assignment de programação: i) vá ao site mostrado abaixo e procure a versão online free do tutorial e no capítulo 1 escolha o knowledge base 1. Vamos exercitar os fatos e regras, a versão mais simples da representação de conhecimento.*

www.learnprolognow.org/lpnpage.php

## Learn Prolog Now!

by Patrick Blackburn, Johan Bos, and Kristina Striegnitz

- LPNI Home
- Free Online Version
- Paperback English
- Paperback Français
- Teaching Prolog
- Prolog Implementations
- Prolog Manuals
- Prolog Links
- Thanks!
- Contact us

[ next ] [ prev ] [ prev-tail ] [ tail ] [ up ]

### Chapter 1 Facts, Rules, and Queries

This chapter has two main goals:

1. To give some simple examples of Prolog programs. This will introduce us to the three basic constructs in Prolog: facts, rules, and queries. It will also introduce us to a number of other themes, like the role of logic in Prolog, and the idea of performing unification with the aid of variables.
2. To begin the systematic study of Prolog by defining terms, atoms, variables and other syntactic concepts.

1.1 [Some Simple Examples](#)

- [Knowledge Base 1](#)
- [Knowledge Base 2](#)
- [Knowledge Base 3](#)
- [Knowledge Base 4](#)
- [Knowledge Base 5](#)

1.2 [Prolog Syntax](#)

- [Atoms](#)
- [Numbers](#)
- [Variables](#)
- [Complex terms](#)

1.3 [Exercises](#)

1.4 [Practical Session](#)

[ next ] [ prev ] [ prev-tail ] [ front ] [ up ]



*Até a próxima aula!*