



Escola Politécnica da USP - Depto. de Enga. Mecatrônica

PMR-3510 Inteligência Artificial

Aula 11- Competição

Prof. José Reinaldo Silva

reinaldo@usp.br





A competição entre equipes

webcid.com.br

Domingo	Segunda	Terça	Quarta	Quinta	Sexta	Sábado
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

21: Início do horário de verão 12: Nsa. Sra. Aparecida 15: Dia dos Professores
02 - Quarto Minguante 09 - Lua Nova 16 - Quarto Crescente 24 - Lua Cheia 31 - Quarto Minguante

No dia 31, segundo o nosso cronograma, teremos o desfecho do trabalho em grupo e faremos isso na forma de uma competição onde todos os grupos deverão resolver o mesmo problema dado como estado inicial. Todos devem ter o seu programa instalado no sistema Swish e pronto para rodar sua implementação do A^* .



Design and Sci... | CEUR-WS.org/V... | ScholarOne Man... | itSIMPLE | Meist... | The service con... | Designing Produ... | IEEE Xplore Full-... | Cloud manufact... | Age of Entangle... | Inbox (270) - re... | Entrada (2.565) | SWISH -- exam...

swish.swi-prolog.org/example/examples.swinb

Bookmarks Bar (Chrom... | Bookmarks | Artificial Intelligence: | Notícias | Popular | Save to Mendeley

SWISH File Edit Examples Help

134 users online Search

Welcome to SWISH

You are reading a SWISH *notebook*. A notebook is a mixture of *text*, *programs* and *queries*. This notebook only contains text and gives an overview of example programs shipped with SWISH.

- **First steps**
 - [Knowledge bases](#) provides a really simple knowledge base with example queries.
 - [Lists](#) defines a couple of really simple list operations and illustrates timing *naive reverse*.
- **Classics**
 - [Movie database](#) provides a couple of thousands of facts about movies for you to query.
 - [Expert system](#) illustrates simple meta-interpretation of rules and asking for missing knowledge.
 - [Eliza](#) implements the classical shrink.
 - [English grammar](#) DCG rules for parsing some simple English sentences and show the result as an SVG tree.
- **Puzzles and constraints**
 - [Einstein's Riddle](#) A famous puzzle attributed to Einstein.
 - [N-Queens \(traditional\)](#) solves the N-queens problem using traditional Prolog and illustrates domain-specific (graphics) output.
 - [N-Queens \(clp\(fd\)\)](#) as above, illustrating the value of constraint programming.
 - [Sudoku \(clp\(fd\)\)](#) solves the sudoku puzzle using constraint programming, redering the result as a table.
 - [Knights and Knaves \(clp\(b\)\)](#) solves boolean problems.
 - [Mortgage \(clp\(q,r\)\)](#) Compute mortgages.
- **Side effects and I/O**
 - [Read and write](#) demonstrates that you can read from and write to the web interface.
 - [Assert and retract](#) demonstrates using the dynamic database.
- **Machine learning (notebooks)** (see also [SWISH tutorials](#))
 - [EM Clustering of the Iris Dataset](#)
 - [SIATEC pattern discovery in polyphonic music](#)
- **Graphical output** (see also [rendering](#))

```
?- trace,in_data(X,Y).
```

Examples History Solutions table results **Run!**



2	8	3
1	6	4
7		5

$$g(0) = 0 + 4$$



1	2	3
8		4
7	6	5

estado final

O estado final será o mesmo que usamos na aula passada no exemplo de resolução da busca baseada no algoritmo Best-first.



O procedimento

- ✦ Todos devem estar preparados com o Swish e o seu algoritmo, mas sem tocar no teclado ou no mouse;
- ✦ O estado inicial será colocado em um slide;
- ✦ Todos esperam o “sinal de largada” para começar o processo;
- ✦ Devem inserir o estado inicial no programa e resolvê-lo;
- ✦ Chegando ao estado final o programa deve descrever este estado e colocar na tela o custo da busca (soma de $g(x)$).
- ✦ com esta informação na saída do programa a equipe faz um sinal de termino e o tempo é anotado.



Após a competição

Após a competição cada equipe deve fazer o upload no sistema e-disciplinas de um texto que descreve a heurística ($g(x)$ e $h(x)$), suas propriedades, e uma listagem do código Prolog. A nota final será a média da nota atribuída a este documento e a nota da competição.



A nota da competição

A nota da competição será a composição da classificação por tempo (de zero a cinco), e cinco pontos se a equipe conseguiu resolver o desafio, mesmo que em um tempo mais longo, e zero se não atingiu o estado final do processo de busca.



Trabalho em grupo

Os grupos estão já definidos e registrados no e-disciplinas. A configuração é a seguinte:

Grupo 1:

Alyson Akio Haro
Alexandre Inoue
Eduardo Kose
Vitor Fukuda

Grupo 2:

Alex Majima
Gabriel Ferreira
Lucas de Angelis
Thiago Ferraz

Grupo 3:

Gabriel Tutia
Lucas Palopoli
Pedro dos Santos Melo
Samuel Monção

Grupo 4:

Gustavo Novello
Luiz Guilherme Sabino
Alessandro Ezequiel Junior
Gabriel Negre

Grupo 5:

Gabriel Yida
Vinicius Santiago
Danilo Polidoro

Grupo 6:

Guilherme Aires de França
Henrique Peterlevitz
Wagner Geraldo Ferreira



O desafio

Como desafio teremos três rodadas onde se colocará um estado inicial e cada um deve buscar atingir o estado final.

1	2	3
8		4
7	6	5

estado final



Primeiro desafio

Todos prontos?

7	2	
1	6	4
8	3	5

estado inicial



Segundo desafio

Todos prontos?

8	4	2
5	3	1
6		7

estado inicial



Terceiro desafio

Todos prontos?

6	7	8
	4	2
3	5	1

estado inicial