



UNIVERSIDADE DE ÉVORA

INTELIGÊNCIA ARTIFICIAL

---

## Relatório do 2º Trabalho Prático

---

*Autor:*

Pedro NUNES, 31240

Tiago MARTINHO, 35735

*Docente:*

Paulo QUARESMA

Abril de 2018

# Índice

1	Pesquisa Local	2
---	----------------	---

# 1 Pesquisa Local

## Definição do problema

Suponha que um labirinto tem dimensão 4x4 e que um agente está no quarto (1,1) e pretende sair do labirinto. O agente não tem conhecimento do local de saída do labirinto nem quais as portas que estão fechadas.

A			O

**Implementação em Prolog de um algoritmo de pesquisa local para ambientes desconhecidos, assumindo que a saída se encontra no quarto (1,4) e que as portas entre os quartos (1,1) e (1,2), (2,1) e (2,2), (3,1) e (4,1), (3,2) e (3,3), e (4,2) e (4,3) estão bloqueadas.**

(Devemos ter em atenção que o local de saída e a deteção de portas bloqueadas só é conhecida pelo agente quando ele entra nos quartos em causa, como tal o agente só é impedido de utilizar uma porta bloqueada caso se encontre numa sala adjacente, como o agente não tem conhecimento do local de saída a função que calcula o valor heurístico segue duas simples regras: valor 1 caso não seja saída e valor 0 caso seja saída.)

```
%no(Estado , no_pai , Operador , Custo , Profundidade)
```

```
pesquisa_local_hill_climbingSemCiclos(E, _) :-  
    retract(visitados(V)),  
    V1 is V + 1,  
    asserta(visitados(V1)),  
    estado_final(E),  
    write(E), write(' '), nl,  
    write("Total de estados visitados: \t"), write(V1), nl.
```

```
pesquisa_local_hill_climbingSemCiclos(E, L) :-  
    write(E), write(' '),  
    expande(E, LSeg),  
    sort(3, @=<, LSeg, LOrd),  
    obtem_no(LOrd, no(ES, Op, _)),  
    \+ member(ES, L),  
    write(Op), nl,  
    (pesquisa_local_hill_climbingSemCiclos(ES, [E|L]) ; write(undo(Op)),  
    write(' '), fail).
```

```

expande(E, L):-
    findall(no(En,Opn, Heur),
            (op(E,Opn,En,-), heur(En, Heur)),

```

## Heurística

```

%heurística
heur([_, (Ycur, Xcur)], 0) :-
    estado_final([_, (Ycur, Xcur)]).
heur([_, (Ycur, Xcur)], 1) :-
    \+estado_final([_, (Ycur, Xcur)]).

```

## Caminho seguido até chegar ao local de saída:

Sul, Sul, Este, Norte, Norte, Este, Este

## Número de quartos visitados no total:

8 quartos