

O algoritmo de conversão de AFND para ER

Grupo: Adriano Pedro
Eduardo Andrade
Fernando De Abreu
Lucas Tejo Sena

18 de Novembro de 2018

Resumo

Mostraremos os algoritmos de conversão de *afnd* para *afd* e de *afd* para *er*.

1 Introdução à idéia geral da conversão

1.1 Explicação teórica da conversão

A conversão consiste em inicialmente utilizar o **estado inicial** e procurar em quais estados pode-se estar a partir desse estado inicial, gerando assim conjunto **P0**.

O algoritmo de *conversão* trabalha percorrendo listas.

2 O algoritmo *afnd* \rightarrow *afd*

O algoritmo de *afnd* para *afd*, consiste na utilização de transições nas listas.

3 Implementação

Para conseguir blablabla

O algoritmo *Minimax* segue abaixo:

Algoritmo 1 Algoritmo Minimax

```
1: function MINIMAX(estado)                                ▷ retorna uma ação
2:   Entradas: estado é a configuração atual do jogo
3:    $v \leftarrow \text{maxvalor}(\text{estado})$ 
4:   retorna a ação  $a$  em sucessores(estado) cujo valor é  $v$ 
5: end function
6: function MAXVALOR(estado)                                ▷ retorna o valor estático
7:   if fim(estado) then
8:     retorna estatico(estado)
9:   end if
10:   $v \leftarrow -\infty$ 
11:  for todas ações  $a$  nos sucessores(estado) do
12:     $v \leftarrow \max(v, \text{minvalor}(a))$ 
13:  end for
14:  retorna  $v$ 
15: end function
16: function MINVALOR(estado)                                ▷ retorna o valor estático
17:   if fim(estado) then
18:     retorna estatico(estado)
19:   end if
20:   $v \leftarrow \infty$ 
21:  for todas ações  $a$  nos sucessores(estado) do
22:     $v \leftarrow \min(v, \text{maxvalor}(a))$ 
23:  end for
24:  retorna  $v$ 
25: end function
```
