UNIVERSIDADE
DE ÉVORA

Inteligência Artificial

Trabalho 2

Miguel Grilo 58387

Jorge Couto 58656

Colégio Luís António Verney

Tabela de Conteúdos

Exercício 1

Alínea a)

- As variáveis representam as posições brancas do tabuleiro 5×5 de Kakuro a preencher com valores entre 1 e 9.
- $v(x(Linha, Coluna), Domínio, ValorAtribuído)$.

Alínea b)

- Cada variável tem como domínio o conjunto de valores de 1 a 9.
- $D = [1,2,3,4,5,6,7,8,9]$

Alínea c)

- Todos os valores numa mesma linha/coluna devem ser diferentes.

$$diff(x(1,3), x(1,4)).\ diff(x(1,3), x(2,3)).\ diff(x(1,3), x(3,3)).$$
$$diff(x(1,4), x(2,4)).\ diff(x(2,1), x(2,2)).\ diff(x(2,1), x(2,3)).$$
$$diff(x(2,1), x(2,4)).\ diff(x(2,1), x(3,1)).\ diff(x(2,1), x(4,1)).$$
$$diff(x(2,2), x(2,3)).\ diff(x(2,2), x(2,4)).\ diff(x(2,2), x(3,2)).$$
$$diff(x(2,2), x(4,2)).\ diff(x(2,3), x(2,4)).\ diff(x(2,3), x(3,3)).$$
$$diff(x(3,1), x(3,2)).\ diff(x(3,1), x(3,3)).\ diff(x(3,1), x(4,1)).$$
$$diff(x(3,2), x(3,3)).\ diff(x(3,2), x(4,2)).\ diff(x(4,1), x(4,2)).$$
$$verifica\_diferentes(Afect):-$$
$$\backslash+\ (\ member(v(Xi, \_, Vi), Afect),$$
$$member(v(Xj, \_, Vj), Afect),$$
$$Xi \backslash= Xj, (diff(Xi, Xj)\ ;\ diff(Xj, Xi)),$$
$$nonvar(Vi), nonvar(Vj), Vi =:= Vj).$$

- As somas parciais devem ser respeitadas.

$$todos\_diferentes([]).$$
$$todos\_diferentes([H|T]:-$$
$$\backslash+member(H, T), todos\_diferentes(T).$$

$$sumlist([], 0).$$

$$sumlist([H|T], Soma):- sumlist(T, Resto), Soma\ is\ H + Resto.$$

$$verifica\_soma(Afect, Vars, Soma):-$$

$$findall(V, (member(X, Vars),$$

$$member(v(X, \_, V), Afect), nonvar(V)), Valores),$$

$$(\ length(Valores, L), length(Vars, L) ->$$
$$sumlist(Valores, Soma), todos\_diferentes(Valores);$$
$$sumlist(Valores, SomaParcial),$$
$$SomaParcial =< Soma, todos\_diferentes(Valores)).$$

$soma\_restricoes([([x(1,3), x(1,4)], 13),$

$([x(1,3), x(2,3), x(3,3)], 24),$

$([x(1,4), x(2,4)], 15),$

$([x(2,1), x(2,2), x(2,3), x(2,4)], 24),$

$([x(3,1), x(3,2), x(3,3)], 23),$

$([x(4,1), x(4,2)], 11),$

$([x(2,1), x(3,1), x(4,1)], 23),$

$([x(2,2), x(3,2), x(4,2)], 9)]).$

$verifica\_somas(Afect) :- soma\_restricoes(Lista),$

$forall(member((Vars, Soma), Lista),$

$verifica\_soma(Afect, Vars, Soma)).$

Alínea d)

- $\%v(x(Linha, Coluna), Domínio, ValorAtribuído)$
  $estado\_inicial(e([v(x(1,3), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(1,4), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(2,1), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(2,2), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(2,3), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(2,4), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(3,1), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(3,2), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(3,3), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(4,1), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(4,2), [1,2,3,4,5,6,7,8,9], \_)], [])).$

Alínea e)

$ve\_restricoes(e(\_, Afect)) :-$

$verifica\_diferentes(Afect),$

$verifica\_somas(Afect).$

Exercício 2

- $\%Backtracking$
  $algoritmo(Metodo) :-$
  $\quad sol(Metodo, Sol, N),$
  $\quad write('Solução: '), write(Sol), nl,$
  $\quad write('Nós\ visitados: '), write(N), nl.$

  $:- dynamic(nos/1).$
  $nos(0).$

  $inc :- retract(nos(N)), N1\ is\ N + 1, asserta(nos(N1)).$

  $backtracking(Sol, N) :-$
  $\quad retractall(nos(\_)), asserta(nos(0)),$
  $\quad estado\_inicial(E0),$
  $\quad back(E0, Sol),$
  $\quad nos(N).$

  $todas\_solucoes\_backtracking(Sol, N) :-$
  $\quad retractall(nos(\_)), asserta(nos(0)),$
  $\quad findall(S, (estado\_inicial(E0), back(E0, S)), Sol),$
  $\quad length(Sol, Total),$
  $\quad Total > 0,$
  $\quad nos(N).$

  $back(e([], A), A).$
  $back(E, Sol) :-$
  $\quad sucessor(E, E1),$
  $\quad inc,$
  $\quad ve\_restricoes(E1),$
  $\quad back(E1, Sol).$

  $sucessor\big(e([v(N, D, V)|R], E), e(R, [v(N, D, V)|E])\big) :- member(V, D).$

  $sol(backtracking, Sol, N) :- backtracking(Sol, N).$

  $sol(todas\_solucoes\_backtracking, Sol, N) :-$
  $\quad todas\_solucoes\_backtracking(Sol, N).$

- Solução Encontrada ($| ?- algoritmo(backtracking).$):

			24	15
	23	13 9	7	6
24	6	1	8	9
23	8	6	9	
11	9	2		

- $| ?- algoritmo(backtracking).$
  Número de Nós Visitados até à primeira solução: 40643
- $| ?- algoritmo(todas\_solucoes\_backtracking).$
  Número de Nós Visitados para todas as soluções: 57204

Exercício 3

- $forward\_checking(Sol, N) :-$
  $retractall(nos(\_)), asserta(nos(0)),$
  $estado\_inicial(E0),$
  $backtracking\_forward(E0, Sol),$
  $nos(N).$

  $todas\_solucoes\_forward(Sol, N) :-$

  $retractall(nos(\_)), asserta\big(nos(0)\big),$

  $findall(S, (estado\_inicial(E0),$

  $backtracking\_forward(E0, S)), Sol),$

  $length(Sol, Total),$

  $Total > 0,$

  $nos(N).$

  $forCheck\big(e(Lni, [v(N, D, V)|Li]), e(Lnii, [v(N, D, V)|Li])\big) :-$
  $corta(N, V, Lni, Lnii).$

$corta(\_, \_, [\,], [\,]).$
$corta(N1, V1, [v(N2, D2, V2)|Resto], [v(N2, D2nova, V2)|RestoNovo]) :-$
$var(V2), (\ diff(N1, N2)\ ;\ diff(N2, N1)\ ),$
$delete(D2, V1, D2nova),$
$D2nova \backslash =\ [\,],$
$corta(N1, V1, Resto, RestoNovo).$
$corta(N1, V1, [v(N2, D2, V2)|Resto], [v(N2, D2, V2)|RestoNovo]) :-$
$\backslash + diff(N1, N2), \backslash + diff(N2, N1),$
$corta(N1, V1, Resto, RestoNovo).$
$backtracking\_forward(e([\,], A), A).$

$backtracking\_forward(E, Sol) :-$
$sucessor(E, E1), inc, ve\_restricoes(E1),$
$forCheck(E1, E2),$
$backtracking\_forward(E2, Sol).$

$sol(forward\_checking, Sol, N) :- forward\_checking(Sol, N).$

$sol(todas\_solucoes\_forward, Sol, N) :-$
$todas\_solucoes\_forward(Sol, N).$

- Solução Encontrada ($| ?- algoritmo(forward\_checking).$):

			24	15
	23	13 9	7	6
24	6	1	8	9
23	8	6	9	
11	9	2		

- $| ?- algoritmo(forward\_checking).$
  Número de Nós Visitados até à primeira solução: 26023
- $| ?- algoritmo(todas\_solucoes\_forward).$
  Número de Nós Visitados para todas as soluções: 36406

Exercício 4

- $melhorado(Sol, N) :-$
  $retractall(nos(\_)), asserta(nos(0)),$
  $estado\_inicial(E0),$
  $backtracking\_melhor(E0, Sol), nos(N).$

  $todas\_solucoes\_melhorado(Sol, N) :-$
  $retractall(nos(\_)), asserta(nos(0)),$
  $findall(S, (estado\_inicial(E0), backtracking\_melhor(E0, S)), Sol),$
  $length(Sol, Total), Total > 0, nos(N).$

  $seleciona\_melhor\_var([H|T], Melhor, Restantes) :-$
  $seleciona\_melhor\_var\_aux(T, H, Melhor, [], Restantes).$

  $seleciona\_melhor\_var\_aux([], Melhor, Melhor, Acc, Acc).$
  $seleciona\_melhor\_var\_aux([H|T], MelhorAtual, Melhor, Acc, Restantes)$
  $:-$
  $H = v(\_, D1, \_), MelhorAtual = v(\_, D2, \_),$
  $length(D1, L1), length(D2, L2),$
  $(L1 < L2 ->$
  $seleciona\_melhor\_var\_aux(T, H, Melhor, [MelhorAtual|Acc],$
  $Restantes);$
  $seleciona\_melhor\_var\_aux(T, MelhorAtual, Melhor, [H|Acc],$
  $Restantes)).$

  $sucessor\_melhor(e(NaoInst, Atrib), e(NovaNaoInst, [v(N, D, V)|Atrib]))$
  $:-$
  $seleciona\_melhor\_var(NaoInst, v(N, D, \_), Resto),$
  $D \backslash= [], member(V, D),$
  $NovaNaoInst = Resto.$

  $backtracking\_melhor(e([], A), A).$
  $backtracking\_melhor(E, Sol) :-$
  $sucessor\_melhor(E, E1), inc,$
  $ve\_restricoes(E1), forCheck(E1, E2),$
  $backtracking\_melhor(E2, Sol).$

$sol(melhorado, Sol, N) :- melhorado(Sol, N).$

$sol(todas\_solucoes\_melhorado, Sol, N) :- todas\_solucoes\_melhorado(Sol, N).$

- Solução Encontrada ($| ?- algoritmo(melhorado).$):

			24	15
	23	13 9	7	6
24	6	1	8	9
23	8	6	9	
11	9	2		

- $| ?- algoritmo(melhorado).$
  Número de Nós Visitados até à primeira solução: 3162
- $| ?- algoritmo(todas\_solucoes\_melhorado).$
  Número de Nós Visitados para todas as soluções: 8734

Exercício 5

Alínea a)

- As variáveis representam as posições brancas do tabuleiro 7 × 7 de Kakuro a preencher com valores entre 1 e 9.
- $v(x(Linha, Coluna), Domínio, ValorAtribuído)$.

Alínea b)

- Cada variável tem como domínio o conjunto de valores de 1 a 9.
- $D = [1,2,3,4,5,6,7,8,9]$

Alínea c)

- Todos os valores numa mesma linha/coluna devem ser diferentes.

$$diff(x(1,1), x(1,2)). diff(x(1,1), x(1,3)). diff(x(1,1), x(2,1)).$$
$$diff(x(1,1), x(4,1)). diff(x(1,1), x(5,1)). diff(x(1,2), x(1,3)).$$
$$diff(x(1,2), x(2,2)). diff(x(1,2), x(3,2)). diff(x(1,2), x(4,2)).$$
$$diff(x(1,2), x(5,2)). diff(x(1,3), x(2,3)). diff(x(1,3), x(3,3)).$$
$$diff(x(1,3), x(4,3)). diff(x(1,3), x(5,3)). diff(x(2,1), x(2,2)).$$
$$diff(x(2,1), x(2,3)). diff(x(2,1), x(2,4)). diff(x(2,1), x(2,5)).$$
$$diff(x(2,1), x(4,1)). diff(x(2,1), x(5,1)). diff(x(2,2), x(2,3)).$$
$$diff(x(2,2), x(2,4)). diff(x(2,2), x(2,5)). diff(x(2,2), x(3,2)).$$
$$diff(x(2,2), x(4,2)). diff(x(2,2), x(5,2)). diff(x(2,3), x(2,4)).$$
$$diff(x(2,3), x(2,5)). diff(x(2,3), x(3,3)). diff(x(2,3), x(4,3)).$$
$$diff(x(2,3), x(5,3)). diff(x(2,4), x(2,5)). diff(x(2,4), x(3,4)).$$
$$diff(x(2,4), x(4,4)). diff(x(2,5), x(3,5)). diff(x(3,2), x(3,3)).$$

$diff(x(3,2), x(3,4)).\ diff(x(3,2), x(3,5)).\ diff(x(3,2), x(4,2)).$
$diff(x(3,2), x(5,2)).\ diff(x(3,3), x(3,4)).\ diff(x(3,3), x(3,5)).$
$diff(x(3,3), x(4,3)).\ diff(x(3,3), x(5,3)).\ diff(x(3,3), x(3,4)).$
$diff(x(3,4), x(3,5)).\ diff(x(3,4), x(4,4)).\ diff(x(4,1), x(4,2)).$
$diff(x(4,1), x(4,3)).\ diff(x(4,1), x(4,4)).\ diff(x(4,1), x(5,1)).$
$diff(x(4,2), x(4,3)).\ diff(x(4,2), x(4,4)).\ diff(x(4,2), x(5,2)).$
$diff(x(4,3), x(4,4)).\ diff(x(4,3), x(5,3)).\ diff(x(5,1), x(5,2)).$
$diff(x(5,1), x(5,3)).$
$verifica\_diferentes(Afect) :-$
$\backslash+\ (\ member(v(Xi, \_, Vi), Afect),$
$member(v(Xj, \_, Vj), Afect),$
$Xi \backslash= Xj, (diff(Xi, Xj)\ ;\ diff(Xj, Xi)),$
$nonvar(Vi), nonvar(Vj), Vi\ =:=\ Vj).$

- As somas parciais devem ser respeitadas.

$todos\_diferentes([]).$
$todos\_diferentes([H|T] :-$
$\backslash+member(H, T), todos\_diferentes(T).$

$sumlist([], 0).$

$sumlist([H|T], Soma) :- sumlist(T, Resto), Soma\ is\ H\ +\ Resto.$

$verifica\_soma(Afect, Vars, Soma) :-$

$findall(V, (member(X, Vars),$

$member(v(X, \_, V), Afect), nonvar(V)), Valores),$

$(\ length(Valores, L), length(Vars, L) ->$
$sumlist(Valores, Soma), todos\_diferentes(Valores);$
$sumlist(Valores, SomaParcial),$
$SomaParcial\ =<\ Soma, todos\_diferentes(Valores)).$

$soma\_restricoes([([x(1,1), x(1,2), x(1,3)], 20),$

$([x(2,1), x(2,2), x(2,3), x(2,4), x(2,5)], 23),$

$([x(3,2), x(3,3), x(3,4), x(3,5)], 14),$

$([x(4,1), x(4,2), x(4,3), x(4,4)], 23),$

$([x(5,1), x(5,2), x(5,3)], 19),$

$([x(1,1), x(2,1)], 13),$

$([x(4,1), x(5,1)], 11),$

$([x(1,2), x(2,2), x(3,2), x(4,2), x(5,2)], 26),$

$([x(1,3), x(2,3), x(3,3), x(4,3), x(5,3)], 28),$

$([x(2,4), x(3,4), x(4,4)], 18),$

$([x(2,5), x(3,5)], 3)]).$

$verifica\_somas(Afect) :- soma\_restricoes(Lista),$

$forall(member((Vars, Soma), Lista),$

$verifica\_soma(Afect, Vars, Soma)).$

Alínea d)

- $\%v(x(Linha, Coluna), Domínio, ValorAtribuído)$
  $estado\_inicial(e([v(x(1,1), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(1,2), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(1,3), [1,2,3,4,5,6,7,8,9],4),$
  $v(x(2,1), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(2,2), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(2,3), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(2,4), [1,2,3,4,5,6,7,8,9],4),$
  $v(x(2,5), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(3,2), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(3,3), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(3,4), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(3,5), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(4,1), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(4,2), [1,2,3,4,5,6,7,8,9],2),$
  $v(x(4,3), [1,2,3,4,5,6,7,8,9],5),$
  $v(x(4,4), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(5,1), [1,2,3,4,5,6,7,8,9], \_),$
  $v(x(5,2), [1,2,3,4,5,6,7,8,9],8),$
  $v(x(5,3), [1,2,3,4,5,6,7,8,9], \_)], [])).$

Alínea e)

$ve\_restricoes(e(\_, Afect)) :-$

$verifica\_diferentes(Afect),$

$verifica\_somas(Afect).$

Alínea f)

- Solução Encontrada (| ? − $algoritmo(backtracking).$):

	13	26	28		
20			4	18	3
23				4	
	14 11				
23		2	5		
19		8			

$Sem\ Solução$

- Solução Encontrada (| ? − $algoritmo(forward\_checking).$):

	13	26	28		
20			4	18	3
23				4	
	14 11				
23		2	5		
19		8			

$Sem\ Solução$

- Solução Encontrada (| ? − $algoritmo(melhorado).$):

	13	26	28		
20			4	18	3
23				4	
	14 11				
23		2	5		
19		8			

$Sem\ Solução$

- Assim, quando executamos os comandos para tentar encontrar uma solução para este Kakuro 7×7, é retornado o output 'no', o que significa que não foi encontrada qualquer solução para este problema após a análise de todas as possibilidades de atribuição, respeitando as restrições definidas. Isto indica que o espaço de pesquisa foi totalmente explorado, e ainda assim não existe nenhuma configuração válida que satisfaça todas as restrições do problema. Neste trabalho foram utilizados três algoritmos para resolver instâncias do problema: Backtracking, onde as variáveis são atribuídas sequencialmente e a cada passo é verificada a consistência com as restrições; Backtracking com forward checking, que além de verificar as restrições atuais, elimina valores dos domínios das variáveis futuras que já não podem ser utilizados; Backtracking com heurísticas, onde foi aplicada a escolha da variável com menor domínio (MRV) para melhorar a eficiência da pesquisa. No entanto, para o exercício 5 (Kakuro 7×7), nenhuma destas abordagens encontrou solução, o que indica que o problema está mal definido ou impossível de satisfazer dadas as suas restrições

Alínea g)

- No exemplo do exercício 5, não existe nenhuma solução possível que satisfaça simultaneamente todas as restrições do problema. Apesar disso, o algoritmo percorre todos os ramos possíveis da árvore de pesquisa, tentando todas as combinações de valores dentro dos domínios definidos. Assim, o Número de nós visitados até à primeira solução é igual ao número total de nós visitados, já que nenhuma solução é encontrada e o Número total de nós visitados é o valor contabilizado com o predicado inc/0 ao longo da execução do algoritmo.
- $|?- algoritmo(metodo).(Exemplo)$
  $Número\ de\ Nós\ Visitados\ até\ à\ primeira\ solução$
  $= Número\ total\ de\ nós\ visitados$
- $|?- algoritmo(todas\_solucoes\_metodo).(Exemplo)$
  $Número\ de\ Nós\ Visitados\ para\ todas\ as\ soluções$
  $= Número\ total\ de\ nós\ visitados$
- $Número\ total\ de\ nós\ visitados$
  $= Valor\ contabilizado\ com\ o\ predicado\ inc/0\ ao\ longo$
  $da\ execução$
- No entanto, não foi possível determinar um Número total de nós visitados em concreto pois essa tentativa não resulta devido a stack overflow.

Exercício 6

- Todo o código utilizado está presente no ficheiro '*kakuro.pl*'.