2º Trabalho de Inteligência Artificial

2018/2019



Resolução do jogo do quadrado mágico e do sudoku como problemas CSP

Docente:  
Irene Pimenta Rodrigues

Realizado por:

Daniel Serrano – 35087

Miguel Serrano - 34149

24 de março de 2019

Conteúdo

[Introdução: 3](#_Toc4505054)

[1º Exercício: 4](#_Toc4505055)

[1.a) 4](#_Toc4505056)

[1.b) 5](#_Toc4505057)

[1.c) 6](#_Toc4505058)

[2º Exercício: 7](#_Toc4505059)

[2.a) 7](#_Toc4505060)

[2.b) 9](#_Toc4505061)

[2.c) 9](#_Toc4505062)

[Conclusão: 10](#_Toc4505063)

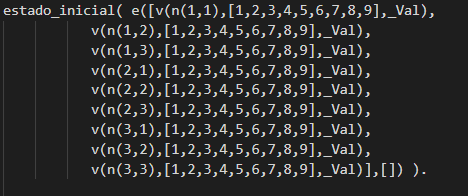
Introdução:

Neste trabalho, no âmbito da cadeira de Inteligência Artificial iremos resolver o jogo do quadrado mágico e do sudoku abordando estes jogos como problemas de satisfação de restrições, e, utilizando as pesquisas de backtrack e forwardcheking.

1º Exercício:

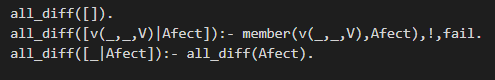
1.a)

Para representar este problema como um problema de satisfação de restrições, considerámos que cada variável tem uma posição, um domínio e o valor da respetiva posição. De referir que o domínio no nosso exemplo vai de 1 a 9 pois o quadrado é de dimensão 3x3 ():

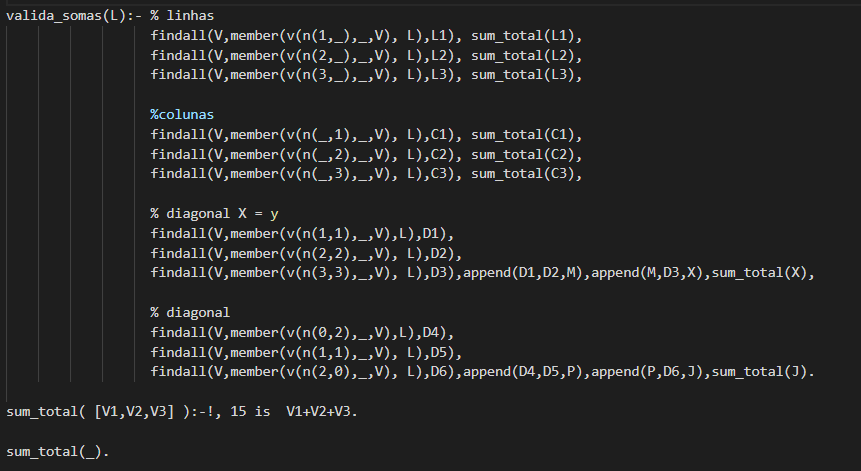


Em relação às restrições o nosso código começa por verificar se todos os elementos do tabuleiro são diferentes:





De seguida verifica as somas de todas as colunas, linhas e diagonais do tabuleiro, comparando essas somas com o número 15 pois é a solução correta de um tabuleiro com a dimensão 3x3:

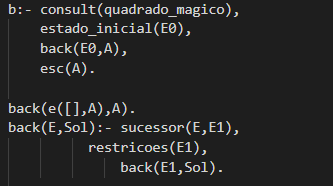


O operador sucessor utilizado foi o mesmo que foi utilizado durante as aulas práticas:

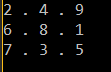
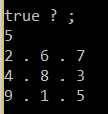
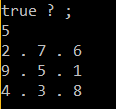
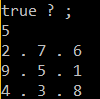


1.b)

Neste exercício era proposto resolver o problema com o algoritmo backtracking, um algoritmo de pesquisa não informada para CSPs e que se revelou bastante eficiente na resolução deste exercício:

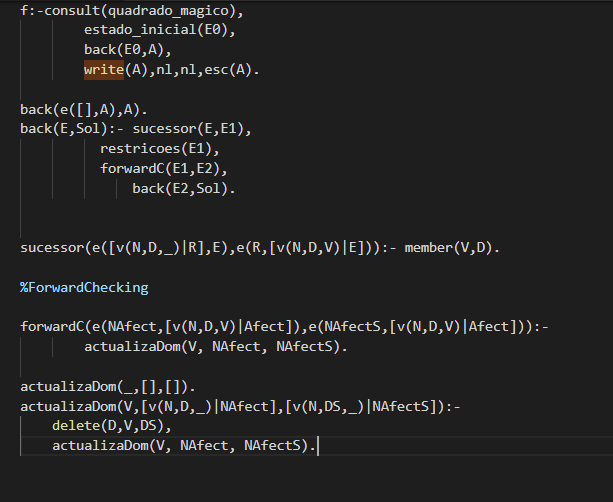


Os resultados obtidos para este algoritmo foram os seguintes:

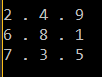
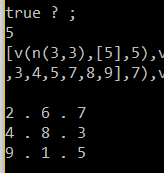
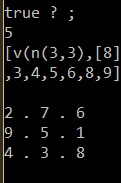
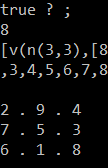
   

1.c)

Neste exercício era proposto resolver o problema com o algoritmo forward checking, um algoritmo que utiliza o algoritmo utilizado no exercício 1.b) mas que o completa verificando certos valores que ainda não foram definidos:



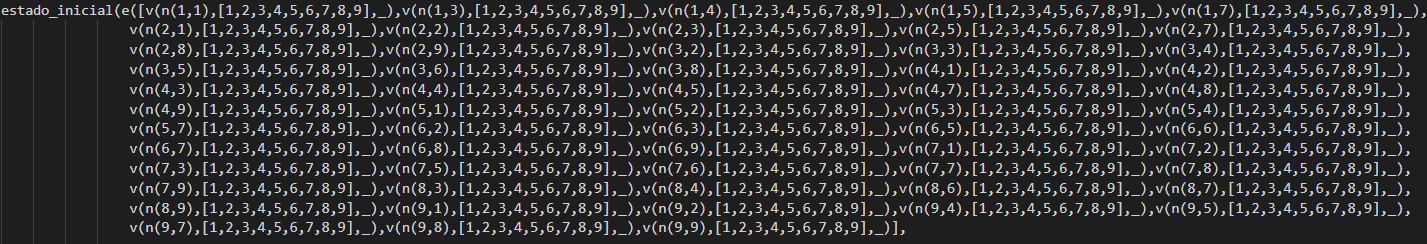
Os resultados obtidos para este algoritmo foram semelhantes ao anterior pois na execução deste jogo a diferença de eficiência não é notória:

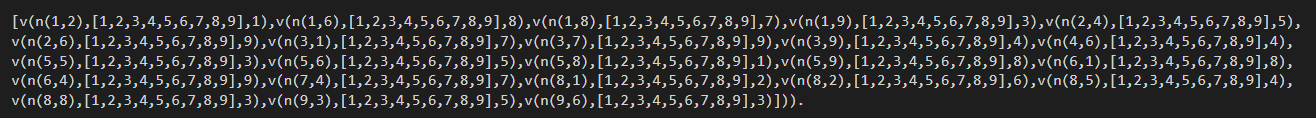
2º Exercício:

2.a)

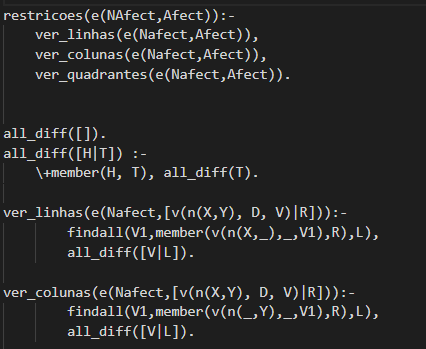
Igualmente ao feito no primeiro exercício (quadrado mágico), iremos representar este problema como um problema de satisfação de restrições com o uso de varáveis que assumem uma posição, um domínio, e o valor da respetiva posição. De referir que neste caso, os valores das posições de cada variável (X,Y) varia entre 1-9 devido à dimensão do tabuleiro ser 9x9 (), e, o domínio das variáveis varia também entre 1-9 pois tratamos quadrantes de dimensão 3x3.

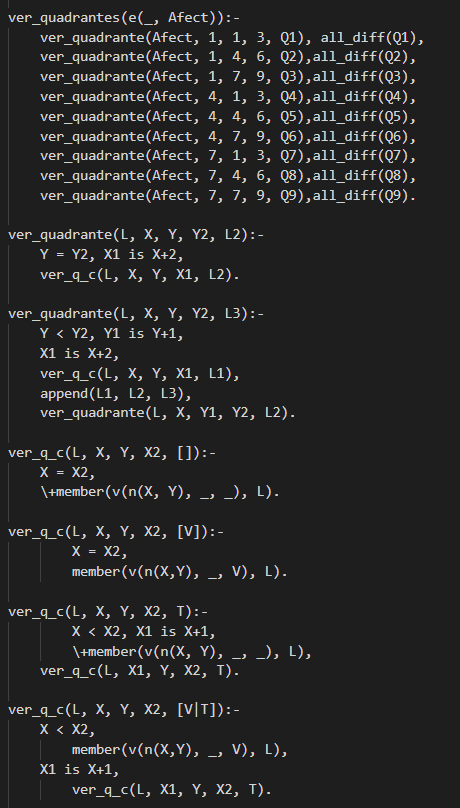
Na imagem seguinte irei apresentar como foram representados os estados iniciais: 

De seguida, apresento as variáveis já preenchidas:



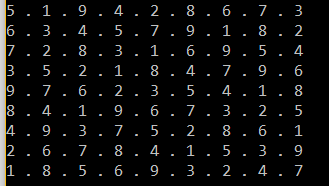
Para representar as restrições, em primeiro foi utilizado um método semelhante ao do exercício anterior em que verificamos as linhas e as colunas, mas, sem a parte em que calculávamos as somas dos valores. Adicionámos ainda, uma restrição denominada “ver\_quadrantes” que verifica se todos os 9 quadrantes, ou seja, “mini” quadrados 3x3 do tabuleiro têm todos os valores diferentes:





2.b)

Neste exercício era proposto resolver o problema com o algoritmo backtracking, um algoritmo de pesquisa não informada para CSPs e que se revelou bastante eficiente na resolução do exercício anterior, mas, neste caso, apesar de não o conseguir comprovar, demorou significativamente mais tempo, no entanto irei apresentar de seguida a resolução obtida:





Como se pode observar com estes resultados, apenas foi possível obter uma resolução para este exercício.

2.c)

Neste exercício era proposto resolver o problema com o algoritmo forward checking, um algoritmo que utiliza o algoritmo utilizado no exercício 1.b) mas que o completa verificando certos valores que ainda não foram definidos, no entanto, ao utilizar o algoritmo funcional do exercício anterior, não conseguis obter nenhum resultado:



Conclusão:

Com a realização deste trabalho ficámos a ter mais conhecimento sobre os tipos de backtracking e forward checking o que, no futuro, penso que estes conhecimentos nos irão dar bastante jeito.  
No entanto, tivemos bastantes dificuldades a implementar o algoritmo de forward checking, principalmente de modo a que funcionasse com o problema do sudoku.