



Departamento de
Sistemas e
Computação

Universidade Federal de Campina Grande
Departamento de Sistemas e Computação
Disciplina: *Inteligência Artificial I*
Prof.: Joseana Macêdo Fachine

Lista de Exercícios N° 07

Aluno: Clerton Ribeiro de Araujo Filho

Obs.: Esta Lista de Exercícios deverá ser respondida por uma equipe composta de, no máximo, 2 alunos.

1. O mapa a seguir apresenta os principais estados e territórios da Austrália. Você tem que atribuir cores a cada região de modo que não haja regiões vizinhas com a mesma cor.



Considere:

Variáveis: WA, NT, Q, NSW, V, SA, T

Domínio: $D_i = \{\text{vermelho, verde, azul}\}$

Restrições: Regiões vizinhas devem ter cores diferentes

Solucione este problema utilizando cada uma das seguintes estratégias:

- a. Retrocesso (*backtracking*);

R: **WA = vermelho;**

NT = vermelho; (falha c/ WA)

NT = verde;

Q = vermelho;

NSW = vermelho; (falha c/ Q)

NSW = verde;

V = vermelho;

SA = vermelho; (falha c/ WA, Q e V);

SA = verde; (falha c/ NT e NSW);

SA = azul;

T = vermelho;

- b. Verificação prévia (*forward checking*);

R: **WA = vermelho;**

=> NT, SA = {verde, azul} (variáveis c/ restrições c/ WA);

=> Q, NSW, V e T = {vermelho, verde, azul};

NT = verde;

=> Q = {vermelho, azul}

=> SA = {azul}

(variáveis c/ restrições c/ NT);

=> NSW, V e T = {vermelho, verde, azul};

Q = vermelho;

=> SA = {azul};

=> NSW = {verde, azul}

(variáveis c/ restrições c/ Q);

=> V e T = {vermelho, verde, azul};

NSW = verde;
 => SA = {azul};
 => V = {vermelho, azul};
 (variáveis c/ restrições c/ NSW);
 => T = {vermelho, verde, azul};
V = vermelho;
 => SA = {azul};
 (variáveis c/ restrições c/ V);
 => T = {vermelho, verde, azul};
SA = azul;
 => T = {vermelho, verde, azul};
T = vermelho;

- c. Heurísticas de variável mais restritiva e mais restringida, respectivamente.

R: Mais restritiva

Candidatas (numero de restrições): WA (2), NT (3), Q(3), NSW (3), V (2), SA (5), T(0);

SA = vermelho;

Candidatas (numero de restrições): WA (2), NT (3), Q(3), NSW (3), V (2), T(0);

NT = verde;

Candidatas (numero de restrições): WA (2), Q(3), NSW (3), V (2), T(0);

Q = azul;

Candidatas (numero de restrições): WA (2), NSW (3), V (2), T(0);

NSW = verde;

Candidatas (numero de restrições): WA (2), V (2), T(0);

WA = azul;

Candidatas (numero de restrições): V (2), T(0);

V = azul;

Candidatas (numero de restrições): T(0);

T = vermelho;

Mais restringido.

Candidatas (cores): WA (verde, vermelho, azul), NT (verde, vermelho, azul), Q (verde, vermelho, azul), NSW (verde, vermelho, azul), V (verde, vermelho, azul), SA (verde, vermelho, azul), T(verde, vermelho, azul);

WA = verde;

Candidatas (cores): NT (vermelho, azul), Q (verde, vermelho, azul), NSW (verde, vermelho, azul), V (verde, vermelho, azul), SA (vermelho, azul), T(verde, vermelho, azul);

NT = vermelho;

Candidatas (cores): Q (verde, azul), NSW (verde, vermelho, azul), V (verde, vermelho, azul), SA (azul), T(verde, vermelho, azul);

SA = azul;

Candidatas (cores): Q (verde), NSW (verde, vermelho), V (verde, vermelho), T(verde, vermelho, azul);

Q = verde;

Candidatas (cores): NSW (vermelho), V (verde, vermelho), T(verde, vermelho, azul);

NSW = vermelho;

Candidatas (cores): V (verde), T(verde, vermelho, azul);

V = verde;

Candidatas (cores): T(verde, vermelho, azul);

T = verde;

2. Eu preciso pagar a minha passagem de moto táxi, que custam 91 centavos. Para pagá-la, eu devo utilizar 5 moedas. O cobrador quer que eu lhe dê uma moeda de 25 centavos ou duas de 10 centavos. Represente isso como um problema de satisfação de restrições e mostre como as heurísticas de propagação de restrições (*forward-checking*) e variável mais restrita agilizam a resolução. Admita que você possui um número infinito de cada moeda.

R: Variáveis: {Moeda 1, Moeda 2, Moeda 3, Moeda 4, Moeda 5};

Domínio: {1, 5, 10, 25, 50};

Restrição: A solução tem que ter uma moeda de 25 centavos, ou duas de 10 centavos, e total deve ser igual a 91 centavos.

Candidatas (centavos): moeda 1 = 25, moeda 2 = {1, 5, 10, 25, 50}, moeda 3 = {1, 5, 10, 25, 50}, moeda 4 = {1, 5, 10, 25, 50}, moeda 5 = {1, 5, 10, 25, 50};

Moeda 1 = 25;

Candidatas (centavos): moeda 2 = {1, 5, 10, 25, 50}, moeda 3 = {1, 5, 10, 25, 50}, moeda 4 = {1, 5, 10, 25, 50}, moeda 5 = {1, 5, 10, 25, 50};

Moeda 2 = 50;

Candidatas (centavos): moeda 3 = 16, moeda 4 = 16, moeda 5 = 16;

Moeda 3 = 10;

Candidatas (centavos): moeda 4 = 6, moeda 5 = 6;

Moeda 4 = 5;

Candidatas (centavos): moeda 5 = 1;

Moeda 5 = 1;

3. Problema: **Alocação de disciplinas**

- **Objetivo:**
 - Todo coordenador tem a cansativa tarefa de montar os horários das disciplinas a cada semestre.
- Admita as seguintes informações para a instituição:
 - tem apenas dois períodos (P1 e P2);
 - tem apenas dois professores (Joseana e Eustáquio);
 - oferece duas disciplinas por período (d1_a, d1_b, d2_a, d2_b);
 - Joseana leciona d1_a e d2_a, e Eustáquio leciona d1_b, d2_b;
 - oferece duas noites de aula por semana, com duas aulas por noite;
 - Joseana não pode ensinar no primeiro horário da segunda;
 - não é permitido que uma mesma disciplina tenha duas aulas por noite.
- Formule uma solução para esse problema:
 - usando CSP, lembrando que você deve estabelecer: (1) Variáveis; (2) Domínio e (3) Regras.
 - a lógica de programação usada (independente de linguagem).
- Implemente a solução para esse problema utilizando a linguagem de programação de sua preferência (**0,5 adicional**).

R: