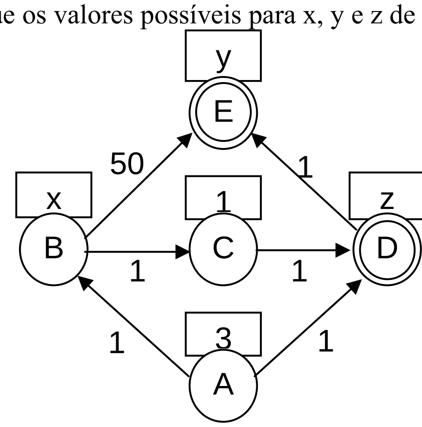
GRUPO I

I.1) Considere o seguinte grafo de estados de um problema de procura. Os valores apresentados nos arcos correspondem ao custo do operador (acção) respectivo, enquanto os valores nos rectângulos correspondem ao valor da heurística. Os estados objectivo são o D e o E. Não se representa os nomes dos operadores, correspondendo cada arco a um operador distinto. Indique os valores possíveis para x, y e z de modo a que a heurística seja consistente.



Resposta:

Para que a heurística seja consistente é necessário que para todo o nó *n* e o seu sucessor *n*' gerado pela acção *a*:

$h(n) \leq c(n,a,n') + h(n')$

No nosso caso isso implicaria que:

$h(A) \leq 1 + h(B)$
$h(A) \leq 1 + h(D)$
$h(B) \leq 1 + h(C)$
$h(B) \leq 50 + h(E)$
$h(C) \leq 1 + h(D)$
$h(D) \leq 1 + h(E)$

ou seja:

$3 \leq 1 + x$
$3 \leq 1 + z$
$x \leq 1 + 1$
$x \leq 50 + y$
$1 \leq 1 + z$
$z \leq 1 + y$

e portanto:

$2 \leq x$
$2 \leq z$
$x \leq 2$
$x \leq 50 + y$
$0 \leq z$
$z \leq 1 + y$

logo:

$x = 2$
$2 \leq z \leq 1 + y$
$y \geq z - 1$

Mas como E e D são estados objectivo, a sua heurística deve ser 0 e por isso não existem valores para x, y e z de modo a que a heurística seja consistente.

I.2) Suponha que os valores de x, y e z na figura da alínea anterior são respectivamente 4, 0 e 100. Especifique a ordem de expansão dos nós efectuada pelo algoritmo A* até atingir uma solução, partindo do estado inicial A.

Resposta:
Seja x = 4, y = 0 e z = 100. Partindo do estado inicial A, os conteúdos das listas aberta e fechada em cada iteração do algoritmo de procura em grafos A* até atingir uma solução são:

Lista Aberta:	A(3)	B(5), D(101)	C(3), E(51), D(101)	E(51), D(101), D(103)
Lista Fechada:		A(3)	A(3), B(5)	A(3), B(5), C(3)

Sendo assim, a ordem de expansão dos nós seria A, B, C até atingir o estado objectivo E.

I.3) Resolva o problema da alínea anterior com um dos seguintes algoritmos de modo a obter a solução óptima: procura sôfrega ou procura de custo uniforme.

Resposta:
Dos algoritmos propostos, o algoritmo de procura de custo uniforme é o único que garante a obtenção da solução óptima. Usando a procura em grafos com o algoritmo de custo uniforme, os conteúdos das listas aberta e fechada até atingir uma solução são:

Lista Aberta:	A(0)	B(1), D(1)	D(1), C(2), E(51)
Lista Fechada:		A(0)	A(0), B(1)

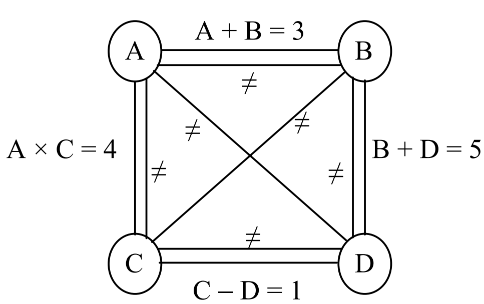
Sendo assim, a solução óptima obtida seria A-D com um custo de 1.

I.4) Considere a seguinte charada numérica em que A, B, C e D são algarismos (entre 1 e 9) todos diferentes entre si. Represente o problema como um problema de satisfação de restrições mostrando o respectivo grafo de restrições e indicando o significado dos nós e das arestas.

A	+	B	=	3
×		+		
C	–	D	=	1
=		=		
4		5		

Resposta:

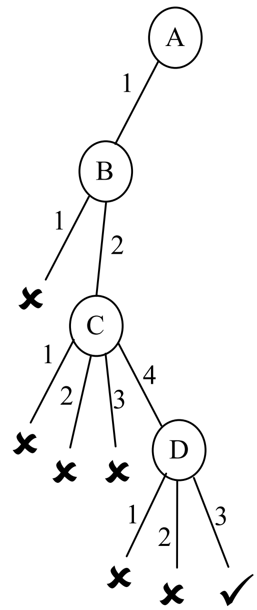
O problema é representado como um problema de satisfação de restrições com o seguinte grafo de restrições em que os nós representam as variáveis e os arcos as restrições:



I.5) Apresente esquematicamente a execução do algoritmo de procura com retrocesso para o problema de restrições da alínea anterior. Use a heurística da variável mais constrangida e desempate pelo número de restrições (e em caso de persistência de empate, por ordem alfabética). Atribua os valores por ordem crescente.

Resposta:

Esquema da execução do algoritmo de procura com retrocesso:



A solução obtida seria então: A = 1, B = 2, C = 4 e D = 3.

I.6) Represente o problema anterior na Linguagem SMODELS. Considere as especificações de domínio **variavel(a;b;c;d)** e **valor(1..9)** e defina uma regra que obrigue a que nos modelos estáveis haja uma e só uma atribuição de um valor a uma variável **atrib(Var,Val)**. Acrescente uma ou mais regras para que as atribuições satisfaçam o conjunto de restrições do problema.

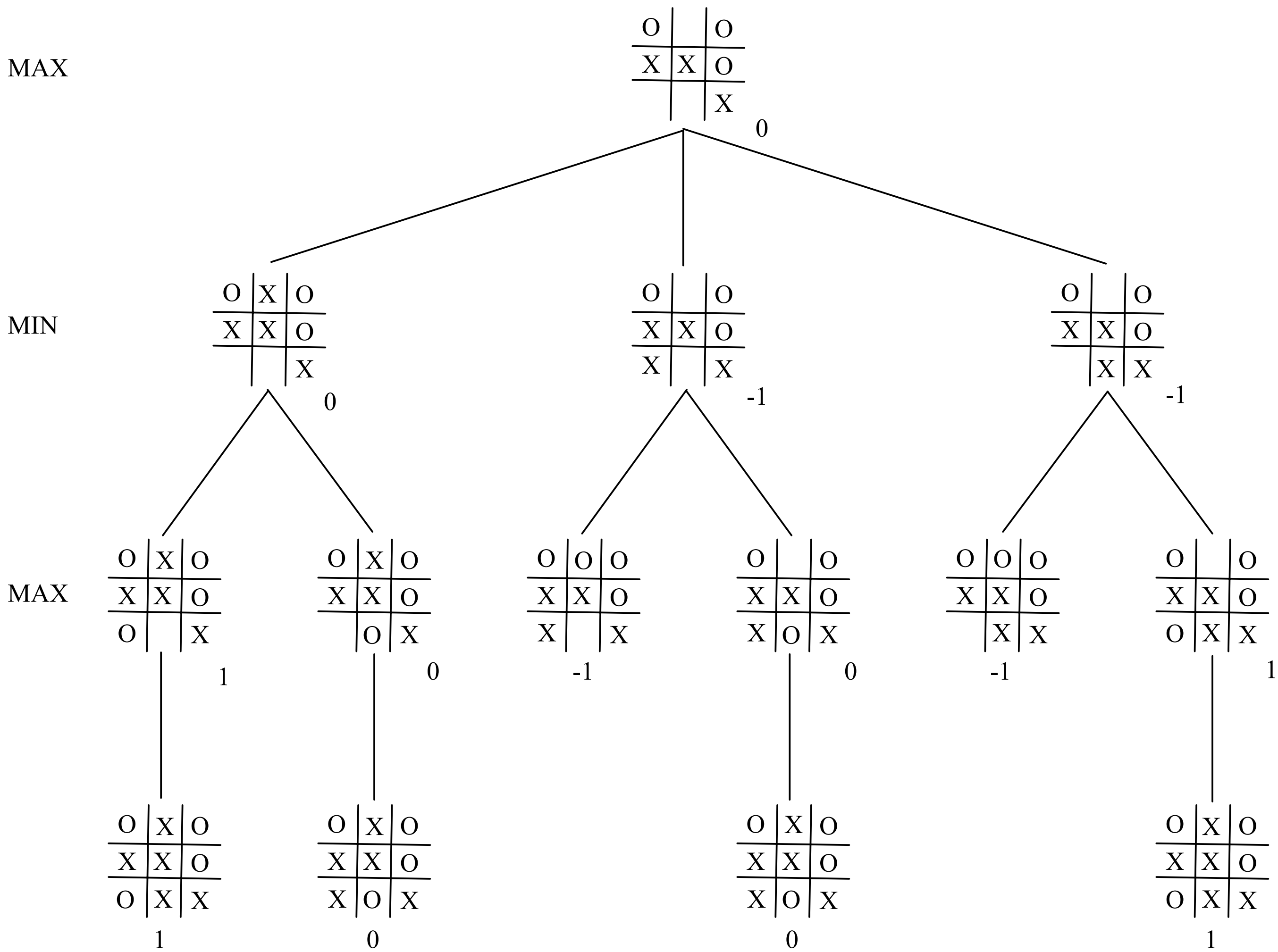
Resposta:

```
valor(1..9).
variavel(a;b;c;d).

1{atrib(V,N):valor(N)}1 :- variavel(V).

:-variavel(V1), variavel(V2), neq(V1,V2), valor(N), atrib(V1,N), atrib(V2,N).

r1 :- valor(N1), valor(N2), atrib(a,N1), atrib(b,N2), eq(N1+N2,3).
:- not r1.

r2 :- valor(N1), valor(N2), atrib(c,N1), atrib(d,N2), eq(N1-N2,1).
:- not r2.

r3 :- valor(N1), valor(N2), atrib(a,N1), atrib(c,N2), eq(N1*N2,4).
:- not r3.

r4 :- valor(N1), valor(N2), atrib(b,N1), atrib(d,N2), eq(N1+N2,5).
:- not r4.
```

I.7) Apresente a árvore de jogo construída pelo algoritmo MINIMAX para o jogo do galo, a partir da posição especificada na figura, do ponto de vista do jogador **X**. Qual deverá ser a próxima jogada de **X**?

O		**O**
X	**X**	**O**
		X

Resposta:



Próxima jogada de X: 1ª linha, 2ª coluna.

I.8) Verifique se $c \wedge e$ é consequência lógica do seguinte conjunto de fórmulas proposicionais, recorrendo ou ao algoritmo de inferência através de tabelas de verdade, ou ao algoritmo de Davis-Putnam. Justifique a resposta.

$a \vee b \vee d$
$b \Rightarrow c$
$b \vee c$
$(c \wedge d) \Rightarrow e$
$d \vee e$

Resposta:
Vamos juntar a negação da conclusão e tentar provar pelo algoritmo de Davis-Putnam que o conjunto de cláusulas obtido é insatisfazível.

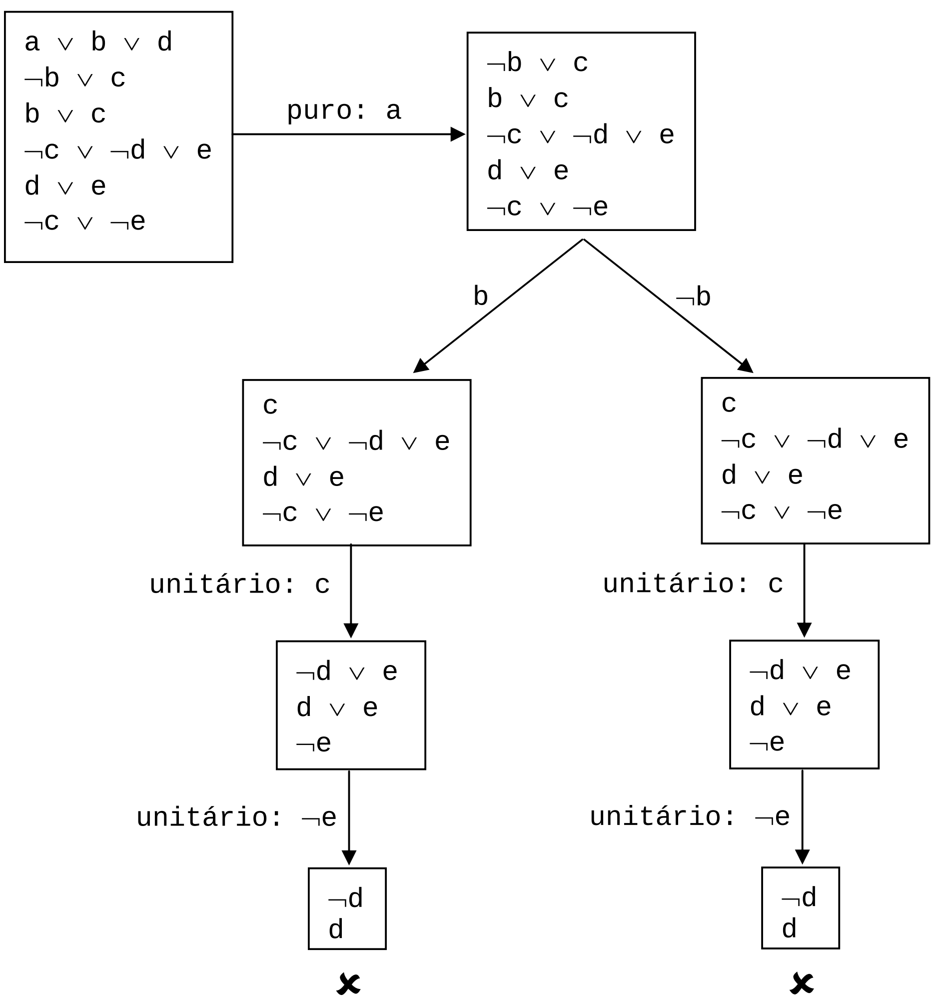
Negação da conclusão: → Forma clausal:

$\neg(c \wedge e) \longrightarrow \neg c \vee \neg e$

Conjunto de fórmulas proposicionais:

Fórmulas		Forma clausal
$a \vee b \vee d$		$a \vee b \vee d$
$b \Rightarrow c$		$\neg b \vee c$
$b \vee c$	→	$b \vee c$
$(c \wedge d) \Rightarrow e$		$\neg c \vee \neg d \vee e$
$d \vee e$		$d \vee e$

Algoritmo de Davis-Putnam:



Como o conjunto de cláusulas é insatisfazível, $c \wedge e$ é consequência lógica do conjunto original de fórmulas proposicionais.

I.9) Seja P o programa em lógica normal listado abaixo. Indique, caso exista, um modelo estável do programa P.

```
a :- b.
b.
c :- b, not d.
e :- c, not f.
a :- e.
a :- f, not b.
```

Resposta:

Seja:

```
I = {a,b,c,e}
```

O programa dividido `P/I` é:

```
a :- b.
b.
c :- b.
e :- c.
a :- e.
```

Cujo modelo mínimo é:

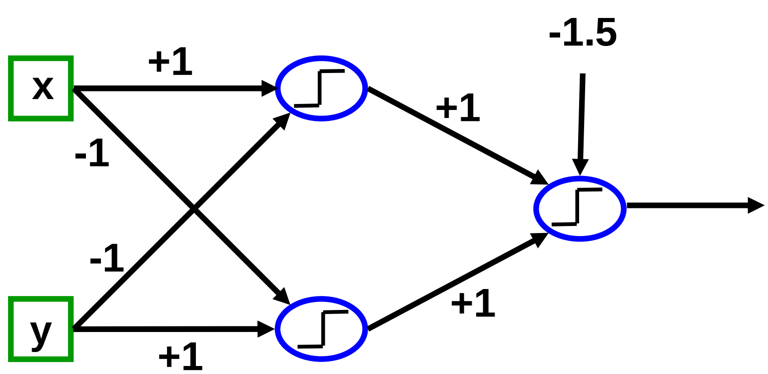
```
least(P/I) = {a,b,c,e}
```

Portanto `I = least(P/I)` logo `I` é um modelo estável.

I.10) Construa uma rede neuronal que implemente a função booleana **x = y** em que x e y são variáveis reais. A rede neuronal deve utilizar apenas neurónios com função de activação limiar (ou degrau). Recorda-se que a função limiar tem o valor 1 quando o seu argumento é maior ou igual a zero; tendo o valor 0 caso contrário.
Sugestão: pense no problema como uma conjunção de duas desigualdades.

Resposta:

$x = y \quad \Leftrightarrow \quad x \leq y \;\wedge\; x \geq y$



GRUPO II

Nos cromossomas humanos, cada gene tem sempre dois alelos, provenientes, de cada um dos progenitores. O gene do grupo sanguíneo tem 3 tipos de alelos A, B e O, podendo portanto aparecer em 6 formas diferentes (AA, AO, AB, BB, BO ou OO). Existem 4 grupos sanguíneos A, B, AB e O. A tabela seguinte descreve a correspondência entre as 6 formas de pares de alelos e o respectivo grupo sanguíneo.

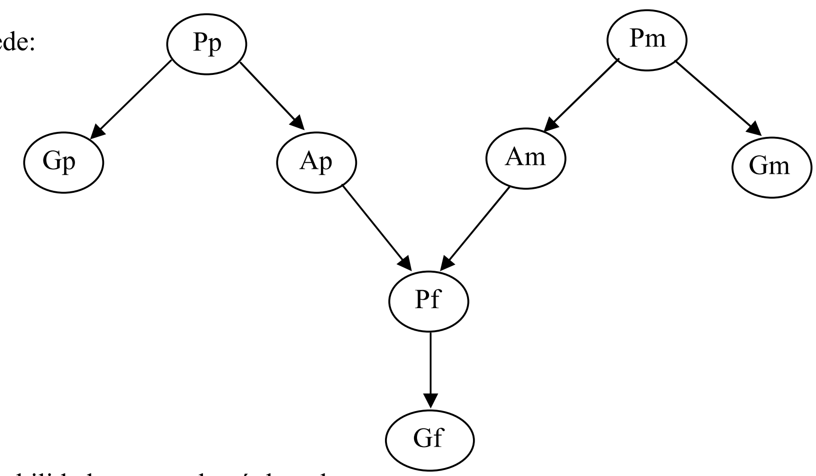
Alelos	AA	AO	AB	BB	BO	OO
Grupo	A	A	AB	B	B	O

Cada progenitor transmite ao seu filho um dos seus alelos, não havendo preferência por nenhum deles. O filho será do grupo correspondente aos alelos recebidos dos seus progenitores (de acordo com a tabela anterior).

II.1) Defina uma rede de Bayes que modele a informação anterior relativamente a 3 indivíduos: o pai, a mãe e o respectivo filho. Para cada individuo deverá ser considerada uma variável que representa o par de alelos (denominada Pp para o pai, Pm para a mãe e Pf para o filho) e uma variável que representa o grupo sanguíneo (denominada Gp para o pai, Gm para a mãe e Gf para o filho). Para os progenitores deve ser ainda considerada uma variável que representa o alelo transmitido ao filho (denominada Ap para o pai e Am para a mãe).
Especifique a topologia da rede e defina as tabelas de probabilidade condicionada. Para atribuir as probabilidades *à priori* das variáveis Pp e Pm considere que a ocorrência de cada alelo é equiprovável nos humanos.

Resposta:

Topologia da rede:



Tabelas de probabilidade para cada nó da rede:

Para o nó Pp (idêntico para Pm):

Pp	P(Pp)
AA	0.111
AO	0.222
AB	0.222
BB	0.111
BO	0.222
OO	0.111

Para o nó Gp (idêntico para Gm e Gf):

<table>
<tr><th rowspan="2">Gp</th><th colspan="6">P(Gp|Pp)</th></tr>
<tr><th>AA</th><th>AO</th><th>AB</th><th>BB</th><th>BO</th><th>OO</th></tr>
<tr><td>A</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>B</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>AB</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>O</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
</table>

Para o nó Ap (idêntico para Am):

<table>
<tr><th rowspan="2">Ap</th><th colspan="6">P(Ap|Pp)</th></tr>
<tr><th>AA</th><th>AO</th><th>AB</th><th>BB</th><th>BO</th><th>OO</th></tr>
<tr><td>A</td><td>1</td><td>0.5</td><td>0.5</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>B</td><td>0</td><td>0</td><td>0.5</td><td>1</td><td>0.5</td><td>0</td></tr>
<tr><td>O</td><td>0</td><td>0.5</td><td>0</td><td>0</td><td>0.5</td><td>1</td></tr>
</table>

Para o nó Pf:

<table>
<tr><th rowspan="3">Pf</th><th colspan="9">P(Pf|Ap,Am)</th><th></th></tr>
<tr><th colspan="3">A</th><th colspan="3">B</th><th colspan="3">O</th><th>Ap</th></tr>
<tr><th>A</th><th>B</th><th>O</th><th>A</th><th>B</th><th>O</th><th>A</th><th>B</th><th>O</th><th>Am</th></tr>
<tr><td>AA</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>AO</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td></td></tr>
<tr><td>AB</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>BB</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>BO</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td></td></tr>
<tr><td>OO</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td></td></tr>
</table>

Nota: use o Netica e consulte o ficheiro GrupoII.neta

II.2) Calcule a probabilidade conjunta do pai ser do grupo sanguíneo A, a mãe ser do grupo sanguíneo B e o filho ser do grupo sanguíneo O.

Resposta:

$$P(Gp = A, Gm = B, Gf = O)$$
$$= \sum_{pp} \sum_{pm} \sum_{ap} \sum_{am} \sum_{pf} P(Pp = pp, Gp = A, Ap = ap, Pm = pm, Gm = B, Am = am, Pf = pf, Gf = O)$$

$$= \sum_{pp} \sum_{pm} \sum_{ap} \sum_{am} \sum_{pf} P(Pp = pp) \times P(Gp = A \mid Pp = pp) \times P(Ap = ap \mid Pp = pp) \times$$
$$P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times P(Am = am \mid Pm = pm) \times$$
$$P(Pf = pf \mid Ap = ap, Am = am) \times P(Gf = O \mid Pf = pf)$$

$$= \sum_{pp} P(Pp = pp) \times P(Gp = A \mid Pp = pp) \times \sum_{pm} P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times$$
$$\sum_{ap} P(Ap = ap \mid Pp = pp) \times \sum_{am} P(Am = am \mid Pm = pm) \times$$
$$\sum_{pf} P(Pf = pf \mid Ap = ap, Am = am) \times P(Gf = O \mid Pf = pf)$$

Como OO é o único valor de Pf para o qual $P(Gf = O \mid Pf = pf) \neq 0$:
$$= \sum_{pp} P(Pp = pp) \times P(Gp = A \mid Pp = pp) \times \sum_{pm} P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times$$
$$\sum_{ap} P(Ap = ap \mid Pp = pp) \times \sum_{am} P(Am = am \mid Pm = pm) \times$$
$$P(Pf = OO \mid Ap = ap, Am = am) \times 1$$

Como O é o único valor de Ap e Am para o qual $P(Pf = OO \mid Ap = ap, Am = am) \neq 0$:
$$= \sum_{pp} P(Pp = pp) \times P(Gp = A \mid Pp = pp) \times P(Ap = O \mid Pp = pp) \times$$
$$\sum_{pm} P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times P(Am = O \mid Pm = pm) \times$$
$$1 \times 1$$

Como AO é o único valor de Pp para o qual $P(Gp = A \mid Pp = pp) \neq 0$ e $P(Ap = O \mid Pp = pp) \neq 0$:
$$= P(Pp = AO) \times 0.5 \times 0.5 \times$$
$$\sum_{pm} P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times P(Am = O \mid Pm = pm) \times$$
$$1 \times 1$$

Como BO é o único valor de Pm para o qual $P(Gm = B \mid Pm = pm) \neq 0$ e $P(Am = O \mid Pm = pm) \neq 0$:
$$= P(Pp = AO) \times 0.5 \times 0.5 \times P(Pm = BO) \times 0.5 \times 0.5 \times 1 \times 1$$

$$= \frac{2}{9} \times \frac{1}{2} \times \frac{1}{2} \times \frac{2}{9} \times \frac{1}{2} \times \frac{1}{2} \times 1 \times 1 = \frac{1}{324} \approx 0.003$$

II.3) Calcule a probabilidade do pai ter transmitido ao filho o alelo O, sabendo-se que a mãe é do grupo sanguíneo B e o filho é do grupo sanguíneo A.

Resposta:

$$P(Ap = O \mid Gm = B, Gf = A) = \frac{P(Ap = O, Gm = B, Gf = A)}{P(Gm = B, Gf = A)} = \alpha \times P(Ap = O, Gm = B, Gf = A)$$

$$= \alpha \times \sum_{pp} \sum_{pm} \sum_{gp} \sum_{am} \sum_{pf} P(Pp = pp, Gp = gp, Ap = O, Pm = pm, Gm = B, Am = am, Pf = pf, Gf = A)$$

$$\begin{aligned}= \alpha \times \sum_{pp} \sum_{pm} \sum_{gp} \sum_{am} \sum_{pf} & P(Pp = pp) \times P(Gp = gp \mid Pp = pp) \times P(Ap = O \mid Pp = pp) \times \\ & P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times P(Am = am \mid Pm = pm) \times \\ & P(Pf = pf \mid Ap = O, Am = am) \times P(Gf = A \mid Pf = pf)\end{aligned}$$

$$\begin{aligned}&= \alpha \times \sum_{pp} P(Pp = pp) \times P(Ap = O \mid Pp = pp) \times \sum_{pm} P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times \\ &\quad \sum_{gp} P(Gp = gp \mid Pp = pp) \times \sum_{am} P(Am = am \mid Pm = pm) \times \\ &\quad \sum_{pf} P(Pf = pf \mid Ap = O, Am = am) \times P(Gf = A \mid Pf = pf)\end{aligned}$$

Como *AA e AO* são os únicos valores de *Pf* para os quais $P(Gf = A \mid Pf = pf) \neq 0$:

$$\begin{aligned}&= \alpha \times \sum_{pp} P(Pp = pp) \times P(Ap = O \mid Pp = pp) \times \sum_{pm} P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times \\ &\quad \sum_{gp} P(Gp = gp \mid Pp = pp) \times \sum_{am} P(Am = am \mid Pm = pm) \times \\ &\quad \left(P(Pf = AA \mid Ap = O, Am = am) \times 1 + P(Pf = AO \mid Ap = O, Am = am) \times 1 \right)\end{aligned}$$

Se *Ap=O* então $P(Pf = AA \mid Ap = O, Am = am) = 0$ e *A* é o único valor de *Am* para o qual $P(Pf = AO \mid Ap = O, Am = am) \neq 0$, logo:

$$\begin{aligned}&= \alpha \times \sum_{pp} P(Pp = pp) \times P(Ap = O \mid Pp = pp) \times \sum_{pm} P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times \\ &\quad \sum_{gp} P(Gp = gp \mid Pp = pp) \times P(Am = A \mid Pm = pm) \times 1\end{aligned}$$

O somatório em *gp* dará sempre 1:

$$\begin{aligned}&= \alpha \times \sum_{pp} P(Pp = pp) \times P(Ap = O \mid Pp = pp) \times \sum_{pm} P(Pm = pm) \times P(Gm = B \mid Pm = pm) \times \\ &\quad P(Am = A \mid Pm = pm)\end{aligned}$$

Como *AA e AO* são os únicos valores de *Pm* para os quais $P(Am = A \mid Pm = pm) \neq 0$ e *BB e BO* são os únicos valores de *Pm* para os quais $P(Gm = B \mid Pm = pm) \neq 0$ podemos concluir que qualquer que seja o valor de *Pm* pelo menos um destes termos será 0:

$$= \alpha \times \sum_{pp} P(Pp = pp) \times P(Ap = O \mid Pp = pp) \times 0 = 0$$

GRUPO III

Considere o problema do caixeiro viajante com N cidades em que, dadas as distâncias entre cada par de cidades, se pretende encontrar o menor circuito que passe por todas as cidades exactamente uma vez. Assuma que as cidades estão numeradas de 1 a N, e que o circuito começa e acaba sempre na cidade número 1.

III.1) Apresente uma expressão matemática para a dimensão do espaço de procura em função do número de cidades N.

Resposta:

Se um estado representar uma sequência de cidades sem repetição (e sem a cidade nº1) então temos:
Estados com 0 cidades: 1
Estados com 1 cidade: N–1
Estados com 2 cidades: (N–1) × (N–2)
Estados com 3 cidades: (N–1) × (N–2) × (N–3)
…
Estados com N–1 cidades: (N–1) × (N–2) × (N–3) × … × 1

Logo, estados com *i* cidades: $\frac{(N-1)!}{(N-1-i)!}$

Portanto a dimensão do espaço de procura será: $\sum_{i=0}^{N-1} \frac{(N-1)!}{(N-1-i)!}$

III.2) Formule claramente o problema para ser resolvido recorrendo a algoritmos de procura em espaço de estados, indicando o estado inicial, teste de estado objectivo e função que devolve os sucessores de um estado, não esquecendo de indicar o custo dos operadores.

Resposta:

Um estado é caracterizado por uma sequência de números (entre 2 e N) que representam a sequência de cidades visitadas no circuito (onde não está representada a cidade nº1 que é a cidade inicial e final).

O estado inicial é uma sequência vazia.

Para testar se um estado é o estado objectivo basta verificar se o número de cidades na sequência é N-1 (a função que devolve os sucessores de um estado garante que todos os elementos da sequência são diferentes entre si).

A função que devolve os sucessores de um estado representado por uma sequência S_0 deverá devolver um sucessor S_i para cada uma das cidades $c_i \in \{2,\ldots,N\}$ que não estejam em S_0 ($c_i \notin S_0$) que será idêntico a S_0 com uma cidade adicional c_i acrescentada no fim da sequência.

Se S_i for o sucessor de S_0 resultante de acrescentar a cidade c_i à sequência S_0, o custo do operador será:

- A distância entre a cidade nº1 e a cidade c_i se S_0 for uma sequência vazia (S_0 é o estado inicial);
- A distância entre a última cidade da sequência S_0 e a cidade c_i, mais a distância entre a cidade nº1 e a cidade c_i, se o número de cidades de S_1 for N-1 (S_1 é um estado objectivo);
- A distância entre a última cidade da sequência S_0 e a cidade c_i se não for nenhum dos casos anteriores;

III.3) Proponha a melhor heurística que consiga, garantindo ao mesmo tempo a obtenção de uma solução óptima pelo algoritmo A* para qualquer dimensão N de problemas desta classe. Justifique adequadamente.

Resposta:

Para garantir a obtenção de uma solução óptima pelo algoritmo A* é necessário que a heurística seja admissível, isto é, que seja optimista. Descreve-se de seguida uma possível heurística admissível.

Seja S um estado que representa um circuito de i cidades, cuja última cidade da sequência é c_i (se S não for uma sequência vazia). Seja Q um conjunto de cidades constituído pelas cidades que não pertencem à sequência de S, mais a cidade c_i (se existir), mais a cidade nº1. Seja v o menor custo entre qualquer par de cidades de Q. Seja l o número de ligações que ainda é necessário incluir no custo do circuito caso S não seja um estado objectivo, isto é, $l = N - i$. Uma heurística admissível devolveria para um nó da pesquisa que contivesse o estado S, o valor 0 se S for um estado objectivo ou o valor $v \times l$ caso contrário.

A heurística anterior é necessariamente optimista, uma vez que atribui o menor custo possível para cada uma das ligações que ainda falta considerar até se chegar a um estado objectivo.

III.4) Indique brevemente como poderia resolver este problema através de um algoritmo de procura local propondo uma função de avaliação e indicando como se poderia obter a vizinhança de um estado.

Resposta:

Para resolver este problema através de um algoritmo de procura local teríamos que considerar um estado como um circuito completo, isto é, corresponderia a um estado objectivo da pergunta III.2).

Uma função de avaliação adequada devolveria o custo total do respectivo circuito, isto é, a soma das distâncias entre as cidades consecutivas do circuito (incluindo a nº1, inicial e final). Assume-se que se pretende minimizar este valor.

Uma vizinhança poderia ser definida por troca entre duas cidades da sequência, isto é, a partir de um estado, poder-se-ia chegar a qualquer outro obtido por troca de duas cidades da sequência.