



Poder Executivo
Ministério da Educação
Universidade Federal do Amazonas
Instituto de Computação - IComp



Disciplina:	Inteligência Artificial	Trab: 2º	Entrega: 23/04.
Professor:	Edjard Mota	Turma: 1	Valor: 30%

Considere que uma partida de dominó de apenas uma rodada, cujo **objetivo é apenas pontuar** mais e ganha quem acabar primeiro suas pedras ou "for o último a bater". Cada jogador não vê as pedras dos demais. A distribuição de pedras é aleatória. Para os exercícios solicitados considerar:

mesa, um termo estruturado `mesa(+F1,+F2,+F3,+F4)`, em que cada F_i é uma lista representando uma fileira de pedras que foram jogadas. O cabeça desta lista representa a última pedra jogada, e portanto a espera de outra jogada. Exemplo:
`mesa([(6,6)],[(6,6)],[(1,6),(6,6)],[(2,6),(6,6)])`.

equipes, é um termo estruturado `equipes(mao(L1),mao(L2),mao(L3),mao(L4))`, em que cada L_i é uma lista de pares (representando) pedras distribuídas ao jogador na posição i , do objeto funcional `esquipes/4`. A função `mao/1` representa que as pedras estão na mão do jogador. Exemplo:

`equipes(mao([(3,3)]),mao([(1,6),(4,4)]),mao([(1,5)]),mao([(0,2)]))`.

Representa uma partida em que, possivelmente, o 1º e 2º jogadores serão os próximos a jogar, e o 3º e 4º jogaram anteriormente, nesta ordem.

estado válido da mesa, definido por `estado(+P1,+P2,+P3,+P4)`, em que cada P_i representa apenas a ponta de uma fileira e pode ser um número entre 0 e 6, ou um par (N,N) em que N é um número entre 0 e 6, com a seguinte interpretação para um valor de P_i :

- Se for um par é porque há uma carroça de N na ponta.
- Se for um número, este representa o valor da última pedra que foi jogada, e que aparece na fila F_i , correspondente da mesa. Sendo N a primeira coordenada do par que esta cabeça de F_i .

Exemplo: `estado(3,0,(5,5),4)`

- a última pedra de F_1 foi um $(3,X_1)$ - terno e algum valor
- a última pedra de F_2 foi um $(0,X_2)$ - nada e algum valor
- a última pedra de F_3 foi um $(5,5)$ - carroça de quina
- a última pedra de F_4 foi um $(4,X_4)$ - quadra e algum valor

Utilize o código fornecido `domino.pl` visto em sala e faça o que se pede.

1. Altere o código para que a variável `E` após a chamada de `distribuiPedras(E)` seja substituída da seguinte forma (valor desta questão 10 %):

```
?- distribuiPedras(E), E = equipes(J1,J2,J3,J4).
E = equipes(mao([(2,3),(1,3),(4,6),(2,4),(0,4),...]), ...)
J1 = mao([(2,3),(1,3),(4,6),(2,4),(0,4),(0,2),(0,0)]),
J2 = mao([(3,5),(4,4),(1,2),(2,5),(3,4),(5,6),(1,5)]),
J3 = mao([(1,6),(1,4),(3,3),(0,3),(2,6),(0,6),(3,6)]),
J4 = mao([(0,1),(6,6),(0,5),(2,2),(4,5),(5,5),(1,1)]).
```

2. Defina um programa `qualEstado/2` que associa uma mesa ao estado válido desta mesa (valor: 10 %). Exemplo:

```
?- qualEstado(mesa([(6,6)],[(3,6),(6,6)],[(5,6),(6,6)],[(6,6)]),E).
E = estado((6,6),3,5,(6,6))
```

Note que a ocorrência de dois pares `(6,6)` indica as duas filas da carroça de sena que foi a primeira pedra a jogar, não valendo 12 mas apenas 6. Esta representação Poderia ser melhorada? Faça uma sugestão.

3. Defina um programa `jogaveis/3` que associa uma mão de pedras, um estado válido da mesa a uma lista da mão que podem jogar neste estado (valor: 15 %). Exemplo:

```
?- jogaveis(mao([(4,4),(1,2),(2,5)]),estado(3,0,(5,5),4),Jogaveis).
Jogaveis = [(4,4),(2,5)]
```

4. Defina um programa `joga/4` que associa uma mão de pedras e uma mesa a uma nova mesa, alterada pela jogada (se possível), de uma pedra da mão, e nova mão resultante (valor: 15 %). Exemplo:

```
M1 = mesa([(6,6)],[(3,6),(6,6)],[(5,6),(6,6)],[(6,6)])
Mao1 = mao([(6,4),(1,2),(2,5),(0,1),(3,4),(2,2)])
?- joga(Mao1,M1,Mao2,M1).
Mao2 = mao([(1,2),(2,5),(0,1),(3,4),(2,2)])
M2 = mesa([(4,6),(6,6)],[(3,6),(6,6)],[(5,6),(6,6)],[(6,6)])
```

5. Considerando e utilizando suas implementações dos itens anteriores:

- (a) Proponha uma definição, usando as estruturas propostas (ou sua), para característica de término de uma partida, e mostre o programa que faz este teste (indicando quais elementos são necessários para isto). (valor: 10 %)
 - (b) Proponha um programa em Prolog, que descreva uma partida que se comporta da seguinte maneira (valor: 30 %):
 - i. Distribui Pedras, e mesa vazia
 - ii. Joga na mesa quem tem carroça de sena, marca a ordem i de quem jogou
 - iii. Enquanto não **terminar o jogo**
 - A. joga na mesa o jogador $(i+1)$ modulo 4¹
 - B. se estado da mesa é igual ao anterior, o jogador $i+1$ modulo 4 passou, se não indique quantos pontos fez e "atualiza" i .

Obs: exibir apenas o estado válido após a jogada e a pontuação, se houver.
6. Faça uma proposta de implementação de funções de utilidade e heurísticas considerando, para cada uma das seguintes alternativas (valor: 10 %):
- (a) apenas o estado válido da mesa e uma mão de pedras de qualquer jogador.
 - (b) o estado válido da mesa e duas mãos de pedras, assumindo que são de parceiros. Por exemplo, jogador 1 e jogador 3, ou jogador 2 e jogador 4. Assuma que a dupla que inicia irá maximizar sua jogada e a outra tentar minimizar.

¹Isso irá garantir que se tenha *round robin*, i.e. depois do último vota a vez para o primeiro.