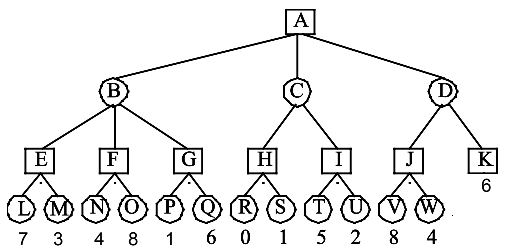
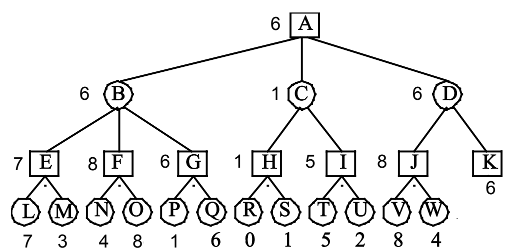
2^o Teste de Inteligência Artificial
7/6/2021
2 horas (17:00 às 19:00)

Grupo I – Considere a árvore da figura acima, que representa o espaço de estados de um jogo de dois jogadores, o valor nas folhas indica o valor da função de utilidade para o estado.

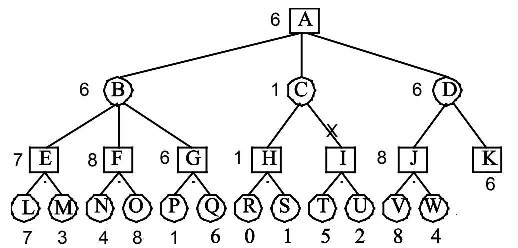


1. Indique o valor dos nós ('A' a 'J') da árvore de acordo com o algoritmo minmax.



Resolução

2. Indique os nós que não precisam de ser avaliados com o corte $\alpha - \beta$.

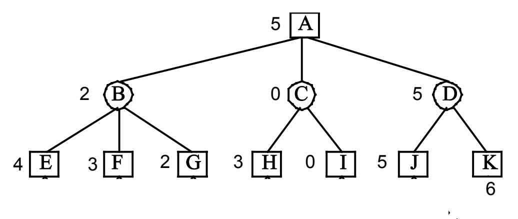


Resolução

I,U,T

3. Indique a jogada perfeita com o minimax e um cuttof= 2 para a seguinte função de avaliação:

   A = 3 B = 5 C =3 D=2 E=4 F=3 G=2 H=3 I=0,J=5,K=1



Resolução

A melhor jogada é a que leva de A a D.

4. Considere o seguinte problema: Um casal de agricultores tem 6 patos e 6 galinhas numa capoeira com capacidade para 6 aves. Com a chegada do inverno decidem que têm que matar 6 aves para colocar na arca. A mulher quer matar os patos todos, mas o marido não está de acordo pois prefere matar as 6 galinhas. Suponha que o casal de agricultores decide jogar o seguinte jogo: colocam os animais em 3 filas de 4.

```
PGPG
GPGP
PPGG
```

Em cada jogada os agricultores podem tirar (matar) entre 1 e n animais iguais, patos ou galinhas, numa fila desde que estejam numa sequência de n animais iguais. O Jogo termina quando tirarem 6 animais e ganha o que ficar com mais animais preferidos.

Considere que a senhora é a primeira a jogar.

(a) Represente o espaço de estados para este jogo e defina os operadores de transição de estados em Prolog.

Resolução Estado – lista com uma lista para cada fila

```
estado_inicial(e([[P,G,P,G],[G,P,G,P],[P,P,G,G]]))

terminal(e(L1,L2,L3)):- conta(L1,N1), conta(L2,N2), conta(L3,N3),
                        N is N1+N2+N3, N=6.

valor(e(L1,L2,L3), 1):- contaG(L1,N1), contaG(L2,N2), contaG(L3,N3),
                        N is N1+N2+N3, N>3.
valor(e(L1,L2,L3), 0):- contaG(L1,N1), contaG(L2,N2), contaG(L3,N3),
                        N is N1+N2+N3, N=3.
valor(e(L1,L2,L3), -1):- contaG(L1,N1), contaG(L2,N2), contaG(L3,N3),
                        N is N1+N2+N3, N<3.

op(e(L1,L2,L3), tira(1,N,A), e(L11,L2,L3)):- tira(L1,N,A,L11),
                                             conta(L11,L2,L3,6).
op(e(L1,L2,L3), tira(2,N,A), e(L1,L22,L3)):- tira(L2,N,A,L22),
                                             conta(L1,L22,L3,6).
op(e(L1,L2,L3), tira(3,N,A), e(L1,L2,L33)):- tira(L3,N,A,L33),
                                             conta(L1,L2,L33,6).



conta(L,N) ---  N é o numero de elementos em L
contaG(L,N) ---  N é o numero de elemento iguais a 'G' em L
conta(A,B,C,N) --- sucede se o a soma dos elemntos em A, B e C é
inferior ou igual a 6.

tira(L,N,A,S) -- retira da lista L N elemntos iguais A se forem
consecutivos. N é >= 1 <= 4.
```

(b) Usando a sua definição de estado e de operadores, desenhe a árvore de minimax com o espaço de estados para decidir a melhor jogada para o jogador A na situação da figura acima. Desenhe a árvore até à profundidade 2, indique o estado em cada nó, e o seu valor se for terminal.

				PG
		GPG		GPGP
		GPGP		PPGG
		PPGG		
				GG
		PPG		GPGP
		GPGP		PPGG
		PPGG		
		PGG		
PGPG		GPGP		...
GPGP		PPGG		
PPGG				
		PGP		
		GPGP		
	12-prof1	PPGG		
		PGPG		
		PGP		
		PPGG		
		PGPG		
		GGP		
		PPGG		
		PGPG		
		GPP		
		PPGG		

Resolução

À profundidade 1 há 12 nós (4 para cada linha) Na profundidade 2 pode haver até 11 nós para cada nó da profundidade 1 Na profundidade 2 não há nós terminais.

Grupo 2 Considere seguinte problema com as torres de Hanoi: tem cinco discos de tamanhos diferentes, três varas, e um robot com dois braços. O robot deve passar os cinco discos da primeira vara para a última vara usando os 2 braços. O robot pode segurar num disco em cada braço, mas numa vara não pode colocar um disco mais largo sobre um mais estreito.

```
      |        |        |              |        |        |
     ---       |        |              |        |        |
    -----      |        |              |        |        |
   -------     |        |           -------     |        |
  ---------    |        |          ---------    |       ---
 -----------   |        |         -----------   |      -----

       Estado inicial                      Estado Final
```

1. Descreva este problema na notação STRIPS. Indique o vocabulário (condições e acções).

Resolução Vocabulário

- Condições: maior(D1,D2) – o disco D1 é maior que o disco D2
- Fluentes:
  namão(D,M) – o disco D está na mão M
  em(D,V) – o disco D está na vara V
  topo(D,V) – o disco D está no topo da vara V
  sobre(D1,D2) – D1 está sobre D2
  livre(M) – a mão M está livre
- Ações:
  agarrar(D,V,M) – agarrar no disco D que está na vara V com a mão M
  PreCond – em(D,V),topo(D,V),livre(M),sobre(D,D1)
  AddL – namao(D,M), topo(D1,V)
  DelL – em(D,V), topo(D,V),livre(M),sobre(D,D1)

  ---

  largar(D,V,M) – largar o disco D que está na mão M na vara V
  PreCond – namao(D,M), em(D1,V), topo(D1,V), maior(D1,D)
  AddL – livre(M), topo(D,V), sobre(D,D1)
  DelL – namao(D,M), em(D,V), topo(D1,V)

2. Represente o estado inicial e o estado final deste problema com o seu vocabulário

```
estado_inicial([maior(d5,d4), ... maior(d5,d1), ...
                maior(d2,d1),
                maior(chao,d5) .... maior(chao,d1)
em(chao,v1), em(chao,v2), em(chao,v3),
em(d1,v1), em(d2,v1), em(d3,v1), em(d4,v1), em(d5,v1),
sobre(d1,d2),sobre(d2,d3),sobre(d3,d4),sobre(d4,d5),sobre(d5,chao),
topo(d1,v1),topo(chao,v2),topo(chao,v3),
livre(esq),livre(dir)
])


estado_final([
em(d1,v3), em(d2,v3), em(d3,v1), em(d4,v1), em(d5,v1),
sobre(d1,d2),sobre(d2,chao),sobre(d3,d4),sobre(d4,d5),sobre(d5,chao),
```

```
topo(d1,v3),topo(chao,V2),topo(d3,v1),
livre(esq),livre(dir)
])
```

3. Indique uma sequência de acções (usando o vocabulário que definiu) para resolver este problema.

   Resolução agarrar(d1,v1,esq),agarrar(d2,v1,dir),largar(d2,v3,dir),largar(d1,v3,esq)

4. Como é que um pop (planeador de ordem parcial) resolveria este problema:

   Indique todos os passos do algoritmo detalhando o plano em cada passo (passos, links e ordem entre passos), indicando quando há ataques quais são as promoções/despromoções se existirem.

   Resolução Coloco mais tarde ....