Universidade Federal do ABC
Bacharelado em Ciência da Computação
Inteligência Artificial - Prof. Denis Gustavo Fantinato
Turma A1 Diurno - Prova B

NOME/RA : ....

Instruções: responda as seguintes questões na folha de prova. Entregue as duas folhas com nome e RA.

Questão 01 (3.0 pontos). Dê uma formulação CSP precisa para o seguinte prob- lema (variáveis X, domínio D, restrições C):

- Alocar N livros em M prateleiras: As prateleiras tem comprimento fixo L. O tamanho de cada livro é diferente. Deve haver pelo menos 10 livros em cada prateleira. Todos os livros devem ser alocados. Cada livro pertence a uma das categorias: 'história', 'matemática', 'geografia', 'programação' ou 'romance'. As categorias 'história', 'geografia' e 'romance' são classificadas como 'Humanas' e 'matemática' e 'programação' são classificadas como 'Exatas'. Uma prateleira pode ter apenas livros de 'humanas' ou 'exatas'.

Variáveis X_i, para $i = 1, ..., N$. Domínio: D = {1,...,M}, pertence à prateleira 1,2, ..., M
Restrições: C: sum(Comprimento({X_i=1})) < L, sum({X_i=1}) > 10, sum(Comprimento({X_i=2})) < L, sum({X_i=2}) > 10, Categoria(X_i) AND Categoria(X_j) in {história,geografia,romance} para i, j = índices de X_k pertencentes a l-ésima prateleira OR [Categoria(X_i) AND Categoria(X_j) in {matematica,programacao} para i, j = índices de X_k pertencentes a l-ésima prateleira]

Questão 02 (4.0 pontos). Deseja-se chegar em Bucharest a partir de Craiova:

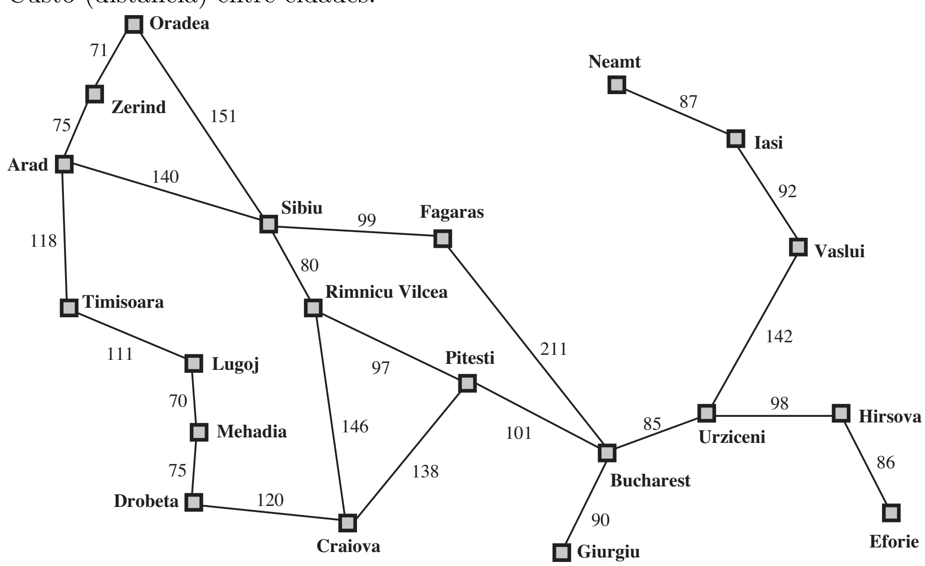
a) Desenhe a árvore com os nós que serão expandidos pelo método de busca em largura até que o destino seja alcançado.

b) Rastreie a operação da busca A* usando a heurística da distância em linha reta. Ou seja, mostre a sequência de nós que o algoritmo irá considerar e o *score* *f*, *g* e *h* para cada nó.

Distância em linha reta até Bucharest:

Arad	366	**Mehadia**	241
Bucharest	0	**Neamt**	234
Craiova	160	**Oradea**	380
Drobeta	242	**Pitesti**	100
Eforie	161	**Rimnicu Vilcea**	193
Fagaras	176	**Sibiu**	253
Giurgiu	77	**Timisoara**	329
Hirsova	151	**Urziceni**	80
Iasi	226	**Vaslui**	199
Lugoj	244	**Zerind**	374

Custo (distância) entre cidades:



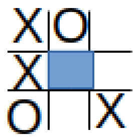
Solução:

a) C
[D R P]
[[M C][S C P][R C B]]]

b) *C[0+160=160]*
D[120+242=362], R[146+193=339], P[138+100=238]
D[120+242=362], R[146+193=339], C[(138+138)+160=436], R[(138+97)+193=428],*B[(138+101)+0=1

Questão 03 (3.0 pontos). Seja um jogo da velha modificado em que a pontuação é dada pela quantidade de pares de elementos (‘X’ ou ‘O’) subjacentes. Por exemplo, para o jogador que insere ‘X’, a pontuação dele será a quantidade de pares consecutivos de ‘X’s na vertical, horizontal ou diagonal. Um dado elemento (’X’ ou ‘O’) pode pertencer a mais de um par. O jogo termina com todo o tabuleiro preenchido.

Considere o jogo com o estado inicial:



O jogador ‘O’ é o próximo a jogar. Responda:

a. Desenhe a árvore Minimax completa para esse jogo e mostre qual é a sequência de nós percorrida caso ambos os jogadores maximizem sua utilidade.
b. Qual será a sequência de nós explorada caso o jogador ‘O’ minimize a utilidade de ‘X’?

Solução:

O: 0[X,O, ; X,N, ; O, ,X;]
X: 0[1[X,O,O; X,N, ; O, ,X;]2[X,O, ; X,N,O; O, ,X;]3[X,O, ; X,N, ; O,O,X;]]
O: 0[1[1[X,O,O; X,N,X; O, ,X;]2[X,O,O; X,N,; O,X,X;]] 2[1[X,O,X; X,N,O; O, ,X;]2[X,O, ; X,N,O; O,X,X;]]3[[X,O,X; X,N, ; O,O,X;],[X,O, ; X,N,X ; O,O,X;]]
Util: 0[1[1[X,O,O; X,N,X; O,O,X;]2[X,O,O; X,N,O; O,X,X;]] 2[1[X,O,X; X,N,O; O,O,X;]2[X,O,O; X,N,O; O,X,X;]]3[[X,O,X; X,N,O; O,O,X;],[X,O,O; X,N,X ; O,O,X;]]
(2,2)(3,3)(1,3)(3,3)(1,3)(2,2)

a) Sequência: 0, 2, 22

b) Sequência: 0, 3, 32

Universidade Federal do ABC
Bacharelado em Ciência da Computação
Inteligência Artificial - Prof. Denis Gustavo Fantinato
Turma A1 Matutino - Prova A

NOME/RA : ....

Instruções: responda as seguintes questões na folha de prova. Entregue as duas folhas com nome e RA.

Questão 01 (3.0 pontos). Compare os algoritmos Hill-Climbing e Simulated Annealing levando em consideração (i) mecanismos de exploração/explotação, (ii) completo/incompleto (iii) convergência para soluções locais. Em quais casos pode ser vantajoso usar o Hill-Climbing? Em quais casos pode ser vantajoso usar o Simulated Annealing?

Questão 02 (2.0 pontos). Explique o que é algoritmo de busca local completo e ótimo.

Questão 03 (5.0 pontos). Uma rede multilayer perceptron (MLP) recebe como entrada amostras de dimensão $\mathcal{R}^{2\times 1}$. A rede possui no total 4 camadas. Cada camada intermediária possui 2 neurônios. A saída da rede é um único valor (escalar). Cada perceptron que compõe a rede possui, além dos pesos sinápticos, um termo de bias e função de ativação relu, que é dada por $max(0, x)$.

a. Desenhe a estrutura dessa rede MLP e forneça as dimensões das matrizes $\mathbf{W}$ que representam os pesos sinápticos dos neurônios nas camadas da rede.

b. Seja o padrão de entrada $\mathbf{x} = [1, 1]^T$. Faça todos os pesos das matrizes $\mathbf{W}$ iguais a 1. Qual é o valor da saída final y?

c. Para fazer a adaptação dos pesos da rede MLP, simule uma iteração da metaheurística Estratégia Evolutiva com parâmetros $ES(\mu/\rho, \lambda) = ES(2/2, 2)$. Considere que a população inicial possui um indivíduo com todos os valores iguais a 1 (como no item anterior) e um indivíduo com todos os valores iguais a 0. Para o cruzamento, ambos serão selecionados. Assuma α = 0,5 para o primeiro cruzamento, α = 0,3 para o segundo cruzamento e também que a operação de mutação gera apenas valores iguais a 0,2 (ou seja, não há aleatoriedade neste caso). Se necessário, para o processo de seleção, use como função objetivo o erro quadrático, considerando o valor desejado igual a 5 para a amostra $\mathbf{x} = [1, 1]^T$.

PROVA 2A

1) i) Hill-Climbing faz apenas explotação atualizando a solução pelo melhor vizinho.

Simulated Annealing permite um equilíbrio entre explorasão e explotasão. No início da execusão, a variável de Temperatura é escolhida alta, permitindo que vizinhos com menor fitness sejam considerados como solusão, expandindo o poder de explorasão. Ao longo das iterasões, a Temperatura é reduzida e apenas os vizinhos melhores são considerados, focando, portanto, na explotasão.

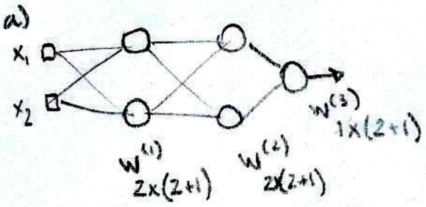
ii) Hill-Climbing é incompleto, pois pode ficar preso em um platô e sequer achar uma solusão para o problema.
Simulated Annealing é completo, pois temperatura permite que se escape de platôs. É quase ótimo, se a temperatura for corretamente ajustada.

iii) Hill-climbing, se não ficar preso em platô, converge para a solusão local mais próxima. Já o Simulated Annealing pode evitar platôs e solusões locais, convergindo para o ótimo global com probabilidade próxima a 1.

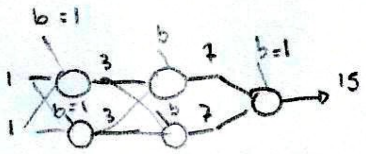
Hill-Climbing é vantajoso em funsões custo unimodais, ou quando se conhece a região onde se encontra o ótimo global. Caso contrário, usar o Simulated Annealing.

2) Algoritmo de Busca Local completo sempre encontra um estado final/objetivo, o ótimo sempre encontra o melhor desses estados.

3) a)



b)



c) ES (2/2,2)

$i_1 = [1\ 1\ \dots\ 1]_{15}$

$i_2 = [0\ 0\ \dots\ 0]_{15}$

cruzamento: $\alpha p_1 + (1-\alpha) p_2$

$f_1 = 0{,}5 i_1 + 0{,}5 i_2 = [0{,}5\ 0{,}5\ \dots\ 0{,}5]_{15}$

$f_2 = 0{,}3 i_1 + 0{,}7 i_2 = [0{,}3\ 0{,}3\ \dots\ 0{,}3]_{15}$

Mutasão f_i + vetor aleatório

$f_1 = f_1 + [0{,}2\ 0{,}2\ \dots\ 0{,}2] = [0{,}7\ 0{,}7\ \dots\ 0{,}7]_{15}$

$f_2 = f_2 + [0{,}2\ 0{,}2\ \dots\ 0{,}2] = [0{,}5\ 0{,}5\ \dots\ 0{,}5]_{15}$

selesão:

Apenas entre os filhos. Manter apenas 2. Como há só 2 filhos,

$i_1 = f_1$

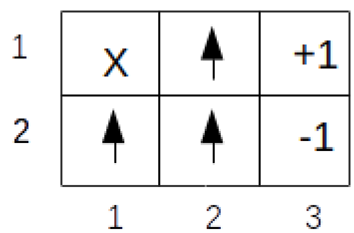
$i_2 = f_2$

Universidade Federal do ABC
Bacharelado em Ciência da Computação
Inteligência Artificial - Prof. Denis Gustavo Fantinato
Turma A1 Matutino - Prova A

NOME/RA : ....

Instruções: responda as seguintes questões na folha de prova. Entregue as duas folhas com nome e RA. É permitido o uso de calculadora.

Questão 01 (3.5 pontos). No contexto de aprendizado passivo, considere a seguinte política fixa e modelo de mundo, com agente iniciando no estado $s = (2, 1)$:



Foram obtidas duas amostras (sequências/tentativas):

$$(2,1)_{-0.1} \mapsto (2,1)_{-0.1} \mapsto (2,2)_{-0.1} \mapsto (1,2)_{-0.1} \mapsto (1,2)_{-0.1} \mapsto (1,2)_{-0.1} \mapsto (1,3)_{+1}$$

$$(2,1)_{-0.1} \mapsto (2,2)_{-0.1} \mapsto (2,1)_{-0.1} \mapsto (2,2)_{-0.1} \mapsto (2,3)_{-1}$$

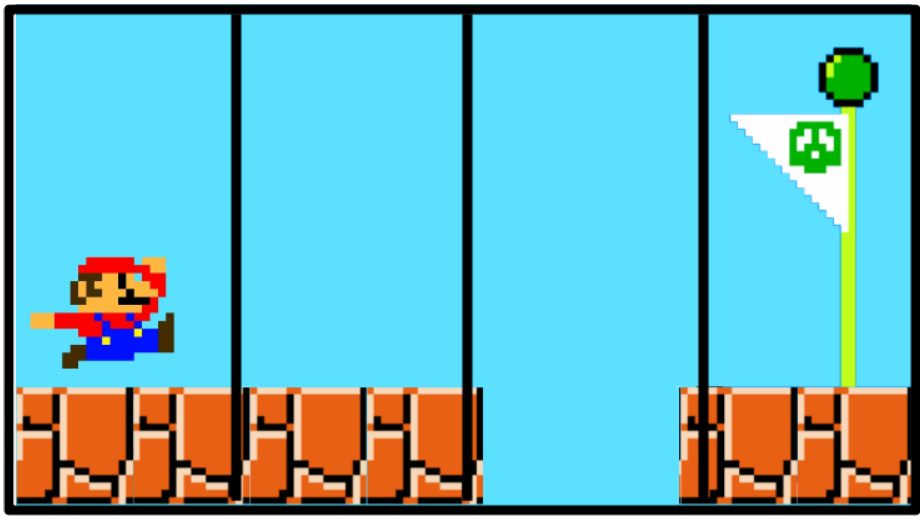
a. Usando a primeira amostra, obtenha a utilidade para a política fixa usando a estimativa direta da utilidade (assuma $\gamma = 1$). Mostre os cálculos envolvidos.

b. Considere a utilidade obtida no item anterior e assuma $\alpha = \gamma = 1$. Usando a segunda amostra, estime os valores da utilidade pelo estimativa da diferença temporal. Mostre os cálculos envolvidos.

Questão 02 (4.0 pontos). No jogo do Super Mario, foi usado um grid de pixels para representar o jogo em quatro estados, conforme a figura da próxima página.

Os estados S_3 e S_4 são estados terminais. As recompensas nos estados S_1, S_2, S_3 e S_4 são -1, -1, -10 e $+10$, respectivamente. O agente (Mario) foi limitado a possuir duas ações: "$\nearrow$" e "$\rightarrow$", que são pular e ir para a frente, respectivamente. Note que, caso o agente pise no estado S_3, ele perde o jogo (e recebe a recompensa -10).

No entanto, durante a programação do código que movimenta o agente, não se considerou corretamente a duração de cada ação na transição de estados e isso gerou um desajuste. A ação "$\nearrow$", por exemplo, faz com que o agente pule mas



S_1 S_2 S_3 S_4

caia no estado seguinte ($S_i \rightarrow S_{i+1}$) em 1/3 dos casos e faz a transição correta, ou seja, pula um estado ($S_i \rightarrow S_{i+2}$), em 2/3 dos casos. Já a ação "$\rightarrow$" tem o comportamento desejado (transição $S_i \rightarrow S_{i+1}$) em 4/5 dos casos, nos 1/5 restantes, o agente faz a transição $S_i \rightarrow S_{i+1} \rightarrow S_{i+2}$ (note que quando o agente passa por S_{i+1}, este pode ser um estado terminal e fazer com que o jogo termine ali).

Assumindo $\gamma = 1$ e a utilidade inicial de cada estado igual às respectivas recompensas, faça uma iteração do algoritmo Iteração-Valor deixando explícito cada passo do algoritmo. Qual a utilidade e a política obtida para cada estado?

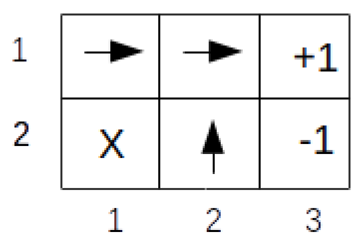
Questão 03 (2.5 pontos). Responda as seguintes perguntas: (*i*) Quais variáveis são assumidas conhecidas em: Processo de Decisão de Markov (MDP) e aprendizado por reforço passivo? (*ii*) Baseado nisso, explique como a equação de Bellman é utilizada no algoritmo Iteração-Valor e no algoritmo Diferença Temporal.

Universidade Federal do ABC
Bacharelado em Ciência da Computação
Inteligência Artificial - Prof. Denis Gustavo Fantinato
Turma A1 Matutino - Prova B

NOME/RA : ....

Instruções: responda as seguintes questões na folha de prova. Entregue as duas folhas com nome e RA. É permitido o uso de calculadora.

Questão 01 (3.5 pontos). No contexto de aprendizado passivo, considere a seguinte política fixa e modelo de mundo, com agente iniciando no estado $s = (1, 1)$:



Foram obtidas duas amostras (sequências/tentativas):

$$(1,1)_{-0.1} \mapsto (1,1)_{-0.1} \mapsto (1,2)_{-0.1} \mapsto (2,2)_{-0.1} \mapsto (1,2)_{-0.1} \mapsto (1,3)_{+1}$$

$$(1,1)_{-0.1} \mapsto (1,2)_{-0.1} \mapsto (2,2)_{-0.1} \mapsto (2,2)_{-0.1} \mapsto (2,3)_{-1}$$

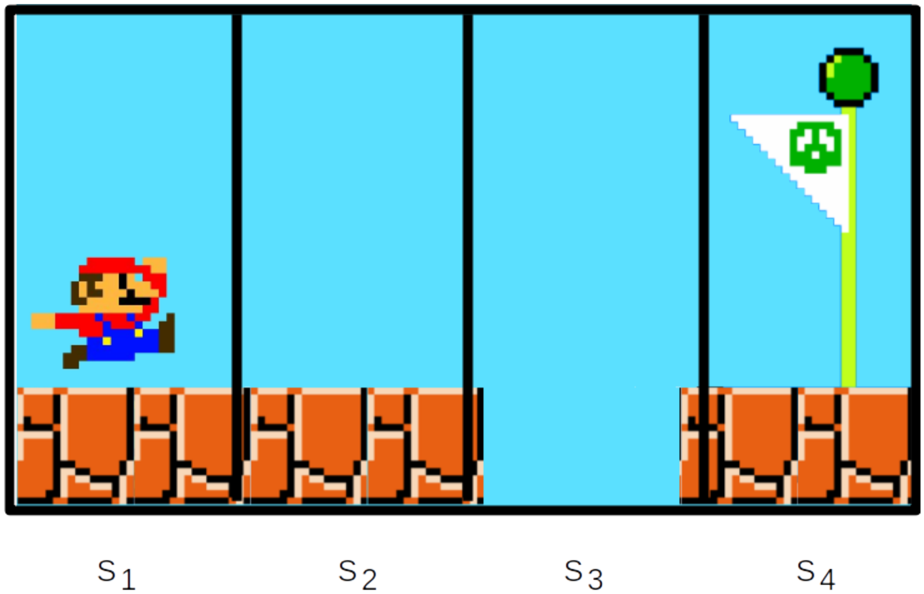
a. Usando a primeira amostra, obtenha a utilidade para a política fixa usando a estimativa direta da utilidade (assuma $\gamma = 1$). Mostre os cálculos envolvidos.

b. Considere a utilidade obtida no item anterior e assuma $\alpha = \gamma = 1$. Usando a segunda amostra, estime os valores da utilidade pelo estimativa da diferença temporal. Mostre os cálculos envolvidos.

Questão 02 (4.0 pontos). No jogo do Super Mario, foi usado um grid de pixels para representar o jogo em quatro estados, conforme a figura da próxima página.

Os estados S_3 e S_4 são estados terminais. As recompensas nos estados S_1, S_2, S_3 e S_4 são -1, -1, -10 e $+10$, respectivamente. O agente (Mario) foi limitado a possuir duas ações: "$\nearrow$" e "$\rightarrow$", que são pular e ir para a frente, respectivamente. Note que, caso o agente pise no estado S_3, ele perde o jogo (e recebe a recompensa -10).

No entanto, durante a programação do código que movimenta o agente, não se considerou corretamente a duração de cada ação na transição de estados e isso gerou um desajuste. A ação "$\nearrow$", por exemplo, faz com que o agente pule mas



caia no estado seguinte ($S_i \rightarrow S_{i+1}$) em 1/3 dos casos e faz a transição correta, ou seja, pula um estado ($S_i \rightarrow S_{i+2}$), em 2/3 dos casos. Já a ação “$\rightarrow$” tem o comportamento desejado (transição $S_i \rightarrow S_{i+1}$) em 4/5 dos casos, nos 1/5 restantes, o agente faz a transição $S_i \rightarrow S_{i+1} \rightarrow S_{i+2}$ (note que quando o agente passa por S_{i+1}, este pode ser um estado terminal e fazer com que o jogo termine ali).

Assumindo $\gamma = 1$, a utilidade inicial de cada estado igual às respectivas recompensas e a política $\pi(S_1) = \rightarrow$ e $\pi(S_2) = \rightarrow$, faça uma iteração do algoritmo Iteração-Política deixando explícito cada passo do algoritmo. Qual a utilidade e a política obtida para cada estado?

Questão 03 (2.5 pontos). Responda as seguintes perguntas: (*i*) Quais variáveis são assumidas conhecidas em: Processo de Decisão de Markov (MDP) e aprendizado por reforço passivo? (*ii*) Baseado nisso, explique como a equação de Bellman é utilizada no algoritmo Iteração-Valor e no algoritmo Diferença Temporal.

GABARITO - PROVA 3A - Matutino

1) a) $U(s) = E\left[\sum \gamma^t R(s)\right]$ (0,25)

(1,0) (0,5)

$U(2,1) = 6 \times (-0{,}1) + 1 = 0{,}4$
$U(2,1) = 5 \times (-0{,}1) + 1 = 0{,}5$ $\Big\}$ $U(2,1) = 0{,}45$

$U(2,2) = 4 \times (-0{,}1) + 1 = 0{,}6$ $\quad U(2,2) = 0{,}6$

$U(1,2) = 3 \times (-0{,}1) + 1 = 0{,}7$
$U(1,2) = 2 \times (-0{,}1) + 1 = 0{,}8$
$U(1,2) = 1 \times (-0{,}1) + 1 = 0{,}9$ $\Big\}$ $U(1,2) = \dfrac{0{,}7 + 0{,}8 + 0{,}9}{3} = 0{,}8$

$U(1,3) = +1$

(0,25)

b) $U(s) = U(s) + \alpha\left(R(s) + \gamma\, U(s') - U(s)\right) = R(s) + U(s')$

(1,5)

- $(2,1) \to (2,2)$

$U(2,1) = 0{,}45 + (-0{,}1 + 0{,}6 - 0{,}45) = 0{,}45 + 0{,}05 = 0{,}5$

- $(2,2) \to (2,1)$

$U(2,2) = 0{,}6 + (-0{,}1 + 0{,}5 - 0{,}6) = 0{,}6 - 0{,}2 = 0{,}4$

- $(2,1) \to (2,2)$

$U(2,1) = 0{,}5 + (-0{,}1 + 0{,}4 - 0{,}5) = 0{,}5 - 0{,}2 = 0{,}3$

- $(2,2) \to (2,3)$

$U(2,2) = -0{,}1 +$ $\begin{cases} U(2,3) = 0 \to U(2,2) = -0{,}1 \\ U(2,3) = -1 \to U(2,2) = -1{,}1 \end{cases}$ considerados corretos

(1,0)

2) $U_{i+1}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a)\, U_i(s')$ $\qquad U_0(s) = [-1, -1, -10, +10]$

It. 1)

$U_1(s_1) = -1 + \begin{cases} \text{"}\nearrow\text{"} & \frac{1}{3} U(s_2) + \frac{2}{3} U(s_3) = \frac{1}{3}(-1) + \frac{2}{3}(-10) = -\frac{21}{3} = -7 \\ \text{"}\rightarrow\text{"} & \frac{4}{5} U(s_2) + \frac{1}{5} U(s_3) = \frac{4}{5}(-1) + \frac{1}{5}(-10) = -\frac{14}{5} = -2{,}8 \end{cases}$

$= -1 - 2{,}8 = -3{,}8$

$U(s_2) = -1 + \begin{cases} \nearrow & \frac{1}{3} U(s_3) + \frac{2}{3} U(s_4) = \frac{1}{3}(-10) + \frac{2}{3}(10) = \frac{10}{3} \\ \rightarrow & U(s_3) = -10 \end{cases}$

$= -1 + \frac{10}{3} = 2{,}33$ (2,0)

U

-3,8	2,33	-10	+10

→	↗		

(1,0)

3) i) MDP: (0,5)

$S, A, R, P, \gamma, s \to s'$

Passivo (0,5)

$\pi, S,$ $\times \to s \to s', P$

ii) It.-Valor: usa Bellman escolhendo a ação imediata que retorna maior U esperada (0,75)

Dif. Temp.: considera uma única transição para definir $s \to s'$ com $p = 1$. (0,75)

GABARITO PROVA B

0,25 Eq.

a) $U(1,1) = 5 \times (-0,1) + 1 = 0,5$
$(1,1) = 0,6$ $\Big\}$ 0,55 (0,5)

$(1,2) = 0,7$ — $U(1,2) = 0,8$
$(2,2) = 0,8$
$(1,2) = 0,9$
$(1,3) = 1$ (1,0)

b) $U(s) = R(s) + U(s')$ (0,25)

(1,5)

$U(1,1) = -0,1 + 0,8 = 0,7$

$U(1,2) = -0,1 + 0,8 = 0,7$

$U(2,2) = -0,1 + 0,8 = 0,7$

$U(2,2) = -0,1 + \to 0 = -0,1$
$\to -1 = -1,1$

2) $U_0 = [-1, -1, -10, +10]$ $\pi_0 = [\to, \to, \times, \times]$

Atualiza U:

$$U(s) = R(s) + \gamma \sum_{s'} P(s' | s, a)\, U(s')$$ (1,0)

$$U(s_1) = -1 + (1)\left[\tfrac{4}{5} \cdot U(s_2) + \tfrac{1}{5} U(s_3)\right] = -1 + \left[-\tfrac{4}{5} - \tfrac{10}{5}\right] = -\tfrac{19}{5}$$

($= -\tfrac{14}{5}$)

$$U(s_2) = -1 + [1 \cdot U(s_3)] = -1 - 10 = -11$$ (2,0)

Verifica π:

$$UE(s_1) = \begin{cases} \to & \tfrac{4}{5} \cdot -11 + \tfrac{1}{5} \cdot -10 = -\tfrac{54}{5} \cong -11 \\ \nearrow & \tfrac{1}{3} \cdot U(s_2) + \tfrac{2}{3} U(s_3) = -\tfrac{11}{3} - \tfrac{20}{3} = -\tfrac{31}{3} \cong -10 \end{cases}$$

$$UE(s_2) = \begin{cases} \to & -10 \\ \nearrow & \tfrac{1}{3} U(s_3) + \tfrac{2}{3} U(s_4) = -\tfrac{10}{3} + \tfrac{20}{3} = \tfrac{10}{3} \checkmark \end{cases}$$

(1,0)

$\pi(s_1) = \nearrow$

$\pi(s_2) = \nearrow$