

Nome: _____

(1,0) Questão 1: Analise as afirmações a seguir sobre agentes e assinale R quando a afirmação for referente a um agente puramente reativo, C quando a afirmação for sobre um agente cognitivo; e A quando a característica se aplicar a ambos.

- a) () Pode ser modelado por meio de máquina de estados e sua prototipação é simples e rápida.
- b) () Usa tradicionalmente arquitetura BDI.
- c) () Sua única forma de comunicação com os demais agentes é indireta, via mudanças no ambiente.
- d) () Seu conhecimento é representado implicitamente nas regras do tipo estímulo-resposta que definem o seu comportamento.
- e) () Alguns dos seus comportamentos podem ser definidos de forma aleatória.
- f) () Consegue trabalhar em domínios parcialmente observáveis e dinâmicos.
- g) () Sua principal característica é a autonomia.
- h) () É capaz de construir planos para resolver problemas.
- i) () Não mantém um histórico das ações realizadas para usar na escolha de ações futuras.
- j) () É capaz de se adaptar a novos ambientes.

(1,5) Questão 2: A versão alfa-beta pruning do Minimax tem melhor desempenho computacional que a versão clássica desse algoritmo, por que ? Fundamente sua resposta (inclua pseudo-código e exemplos, se julgar necessário).

(1,5) Questão 3: Por que o algoritmo Simulated Annealing tem mais chances de achar a solução ótima que a versão clássica do Hill Climbing ? Justifique.

(2,0) Questão 4: Um projeto possui n tarefas que precisam ser distribuídas entre 3 pessoas da equipe. Para cada tarefa foi **estimado o número de horas** necessário para concluí-la. Para responder aos itens a seguir, considere as tarefas e sua carga (em horas) abaixo e que, na solução desse problema, é usado o algoritmo genético.

t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
5	10	15	3	10	5	2	16	9	7

- a) Como as soluções poderíamos ser codificadas para representar as soluções desse problema ? Forneça um exemplo. (0,5)
- b) Defina uma população inicial de tamanho 5. (0,5)
- c) Defina uma função de aptidão para avaliar os cromossomos e exemplifique o seu uso. (0,5)
- d) Mostre um exemplo de cruzamento para cromossomos dessa população. (0,5)

(2,0) Questão 5: Um agente Aspirador de Pó é responsável pela limpeza de quatro salas. Duas dessas salas estão localizadas no térreo e as outras duas, no segundo andar de um prédio. O acesso do agente às salas é sempre a partir do corredor. Considere ainda que esse agente possui o seguinte conjunto de operações:

- *Limpar (S)*: aspira a sala S em que está posicionado.
- *Entrar (C,A,S)*: entra em uma sala S desde que esteja no corredor C do andar A em que essa sala se localiza.
- *Sair(S,A,C)*: sai para o corredor C de um andar A, desde que esteja em uma sala S daquele andar.
- *Subir(C1,C2)*: sobe para o corredor C2 do segundo andar desde que esteja no corredor C1 do

térreo.

- *Descer(C2,C1)*: desce para o corredor C1 do térreo desde que esteja no corredor C2 do segundo andar.

Com base nas descrições dadas, defina:

- (1,6) As operações *Subir* e *Entrar* em STRIPS .
- (0,4) O Plano necessário que permite o agente a partir do estado inicial atingir o estado meta, onde:

Initial state: Em(Agente,S2), Estado(S1,Suja), Estado(S2,Limpa), Estado(S3,Suja), Estado(S4,Limpa), Sala(S1), Sala(S2), Sala(S3), Sala(S4), Andar(S1,Térreo), Andar(S2,Térreo), Andar(S3,Segundo), Andar(S4,Segundo), Andar(C1,Térreo), Andar(C2,Segundo), Corredor(C1), Corredor(C2)

Goal state: Estado(S1,Limpa), Estado(S2,Limpa), Estado(S3,Limpa), Estado(S4,Limpa)

(2,0) Questão 6: Relacione a coluna da esquerda com a da direita, associando os algoritmos de busca com informação às suas características. Ao responder, considere apenas as versões clássicas desses algoritmos.

- | | |
|----------------------------------|--|
| (1) A* | () algoritmo de refinamento que usa uma escala de temperatura para controlar sua execução. |
| (2) <i>Simulated Annealing</i> | () algoritmo de refinamento que move-se continuamente e somente na direção de uma solução melhor, examinando apenas os estados vizinhos imediatos à solução atual. |
| (3) <i>Hill Climbing</i> | () algoritmo que combinado com o Dijkstra faz busca em profundidade e em amplitude, e utiliza função heurística. |
| (4) Algoritmo Genético | () algoritmo de busca adversária. |
| (5) Minimax | () algoritmo que trabalha em cima de um conjunto de soluções candidatas (população) e procura refiná-las geração a geração. |
| | () algoritmo que, no início da sua execução, em menos 1% dos casos aceita como solução, soluções vizinhas piores que a melhor solução encontrada até aquele momento. |
| | () algoritmo de refinamento em que a solução sucessoras à atual é gerada aleatoriamente. |
| | () algoritmo que mistura (cruza) boas soluções para gerar novas soluções. |
| | () algoritmo do tipo <i>Zero-Sum Game</i> em que tenta maximizar as suas vantagens de sucesso, minimizando as vantagens do oponente. |
| | () algoritmo que usa como um dos critérios de parada a convergência, a qual acontece quando mais de 90% da população de soluções candidatas representa a mesma solução. |