Examen intermedio

Inteligencia artificial, período 2017–2.

Profesor: Julio Waissman Vilanova.

Nombre: ____

1. (20 puntos) Responde a los siguientes enunciados como falso o verdadero.

   (a) ___ Existen dos entornos E_1 y E_2 completamente diferentes, en el cual un agente A es perfectamente racional en ambos.

   (b) ___ Existe al menos un entrono E para el cual el agente aleatorio es perfectamente racional.

   (c) ___ Todos los entornos tienen *al menos* un agente perfectamente racional.

   (d) ___ Un agente que juega dominó por parejas se desempeña en un entorno dinámico, discreto, parcialmente observable y estocástico.

   (e) ___ Es posible asegurar que, bajo ciertas condiciones, el algoritmo de temple (recocido) simulado siempre va a encontrar un mínimo global.

   (f) ___ Si tenemos un algoritmo genético con un solo individuo, el algoritmo se reduce a una búsqueda por descenso de colinas.

   (g) ___ Para los problemas CSP con restricciones binarias, mientras mayor sea el grado del método de revisión de consistencia (0-consistencia, 1-consistencia, 2-consistencia, ...) menor el número de *backtrackings* y por lo tanto el algoritmo tomará menos tiempo en encontrar una solución.

   (h) ___ El algoritmo de búsqueda A^* siempre revisa menos nodos que el algoritmo de búsqueda a profundidad (DFS).

   (i) ___ Si $h_1(n)$ y $h_2(n)$ son dos heurísticas admisibles, entonces $h_3(n) = (h_1(n) + h_2(n))/2$ tambien es una heurística admisible.

   (j) ___ El método de búsqueda por poda α–β es un método de búsqueda primero en profundidad, por lo que su problema es el tiempo de ejecución y no la cantidad de memoria utilizada.

2. (20 puntos) Supongamos que tenemos un problema de minimización en un espacio con 10^{12} estados diferentes (los estados tienen 12 variables con 10 valores cada una). Del problema conocemos la función de costo y tenemos una función para generar la lista de vecinos de cada estado $vecinos(x)$, y por supuesto, podemos tener una función $vecino_A(x)$ que devuelve un vecino de el estado x en forma aleatoria.

   (a) (5 puntos) Decidimos resolver el problema con descenso de colinas con reinicios aleatorios, y para esto se realizaron 1000 descensos de colinas con estado inicial aleatorio. los costos de los estados iniciales fue en promedio de 9.1 (mínimo de 6.3 y máximo de 11.3). Se encontró que el valor mínimo encontrado fue 1.3, el máximo de 4.9, y las soluciones en promedio dieron 3.2. En promedio en 5 pasos se encontraba el valor final en cada búsqueda individual.

¿Podríamos decidir quedarnos con el valor de 1.3 como un mínimo global? Justifica tu respuesta:

(b) (5 puntos) Para validar el resultado decidimos utilizar otro método de optimización, digamos un método de temple (recocido) simulado. Para que el método funcione, decidimos utilizar una temperatura inicial de ______________ y un método de calendarización dado por la formula ______________.

Completa la tabla de abajo con la probabilidad de pasar de un estado a un estado vecino, de acuerdo al algoritmo tal como se ejecutó.

Costo estado	Costo vecino(estado)	iteración	Temperatura	probabilidad cambio
4.3	4.5	2		
4.3	4.5	20		
4.3	4.2	3		
4.3	4.2	15		
5.3	10.2	1		
5.3	10.2	5		

(c) (5 puntos) Cuauhtemoc llegó más tarde, y nos avisa que nos dieron mal la información, y que la función de costo $c(x)$ no es la correcta. Marca en cuales casos el resultado encontrado, también sería el resultado correcto para la nueva función de costo $c'(x)$:
□ $c'(x) = 30 \log_2(c(x) + 10)$ □ $c'(x) = 100 \exp(c(x))$ □ $c'(x) = 100 \exp(-c(x))$
□ $c'(x) = c(x)(1 - c(x))$ □ $c'(x) = 0.2c(x) - 100$

(d) (5 puntos) Supongamos que ahora lo que queremos es resolver el mismo problema con un algoritmo genético, por lo que necesitamos establecer cuales serán los operadores genéticos más importantes. Completa las frases y jutifica tus respuestas.

1. La población tiene un tamaño de ______________debido a
2. La adaptación se calcula como ______________debido a
3. El operador de cruza se seleccionó como ______________debido a

4. Se seleccionó ______________como operador de mutación debido a

____

____

5. Se impuso el *elitismo* debido a

____

____

3. (20 puntos) el Pato Donald (PD) se va a vivir con Daisy (D), sus sobrinos Hugo (H), Paco (P) y Luis (L) y su perro Plinche (Pl) en una casa nueva con 6 cuartos. Como les gusta complicarse la vida, numeraron los cuartos (del 1 al 6) y establecieron las siguientes reglas para vivir juntos:

   - Cada uno tiene su propio cuarto.
   - El número del cuarto del Pato Donald es mayor que 3.
   - El número del cuarto de Paco es menor que el del Pato Donald.
   - El número del cuarto de Daisy es 5 o 6.
   - El número del cuarto del Pato Donald es mayor que el de Daisy
   - El cuarto de Hugo tiene número par
   - Luis no tiene ni el primero, ni el último de los cuartos.
   - Luis tiene un cuarto contiguo al de Plinche (la diferencia entre ambos números es 1)
   - Paco y el Pato Donald están separados por un cuarto (la diferencia entre ambos números es 2).

   (a) ($2\frac{1}{2}$ puntos) El problema se puede establecer como un gráfo de restricciones con ______________ variables, las cuales tienen ______________ restricciones unarias y ______________ restricciones binarias.

   (b) (5 puntos) De la tabla siguiente, elimina los valores del domino de cada variable que se pueden reducir por las restricciones unarias.

PD	1	2	3	4	5	6
D	1	2	3	4	5	6
H	1	2	3	4	5	6
P	1	2	3	4	5	6
L	1	2	3	4	5	6
Pl	1	2	3	4	5	6

   (c) ($2\frac{1}{2}$ puntos) De acuerdo a los criterios vistos en clase, ¿Cual sería la primer variable seleccionada? □ PD □ D □ H □ P □ L □ Pl

   (d) (5 puntos) Si asumimos que la primer variable seleccionada fue PD, y se asignó un valor de 6, realiza la reducción de dominios (despues de las restricciones unarias) aplicando 1-consistencia (*forward checking*)

PD	1	2	3	4	5	6
D	1	2	3	4	5	6
H	1	2	3	4	5	6
P	1	2	3	4	5	6
L	1	2	3	4	5	6
Pl	1	2	3	4	5	6

(e) (5 puntos) Ahora vamos a suponer que decidimos solucionar el problema por *mínimos conflictos*. Si consideramos una asignación inicial dada por {PD: 6, H: 4, Pl: 3, P: 2, L: 1, D: 5}, rellena en la tabla de abajo el número de conflictos de cada variables y marca cual es el valor que se le asignaría si se seleccionara para ser modificada dicha variable por el algoritmo.

Variables	Conflictos	1	2	3	4	5	6
PD							
D							
H							
P							
L							
Pl							

4. (20 puntos) En esta sección vamos a explorar que pasa si modificamos el algoritmo de búsqueda por costo uniforme (UCS), modificando los costos locales por la variable $d_{i,j}$.

   Para estas preguntas vamos a considerar solamente búsquedas en árboles (sin guardar estados visitados), donde $c_{i,j} > 0$ es el costo local entre el nodo padre i y el nodo sucesor j. Se asume que solamente tenemos un solo estado meta, y que los sucesores siempre se expanden en orden alfabético. Todas las heurísticas se consideran admisibles.

   Vamos a considerar que dos métodos de búsueda son *equivalentes* en este caso, si ambos métodos expanden los mismos nodos y en el mismo orden.

   (a) (5 puntos) Selecciona todas las opciones en las cuales la selección de $d_{i,j}$ haga que la búsqueda UCS sea equivalente a la búsqueda primero a lo ancho (BFS).

   - ☐ $d_{i,j} = 0$
   - ☐ $d_{i,j} = \alpha$, donde $\alpha > 0$
   - ☐ $d_{i,j} = \alpha$, donde $\alpha < 0$
   - ☐ $d_{i,j} = 1$
   - ☐ $d_{i,j} = -1$
   - ☐ Ninguno de los anteriores

   (b) (5 puntos) Selecciona todas las opciones en las cuales la selección de $d_{i,j}$ haga que la búsqueda UCS sea equivalente a la búsqueda primero a lo profundo (DFS).

   - ☐ $d_{i,j} = 0$
   - ☐ $d_{i,j} = \alpha$, donde $\alpha > 0$
   - ☐ $d_{i,j} = \alpha$, donde $\alpha < 0$
   - ☐ $d_{i,j} = 1$
   - ☐ $d_{i,j} = -1$
   - ☐ Ninguno de los anteriores

   (c) (5 puntos) Selecciona todas las opciones en las cuales la selección de $d_{i,j}$ haga que la búsqueda UCS sea equivalente a la búsqueda UCS original con $d_{i,j} = c_{i,j}$.

   - ☐ $d_{i,j} = c_{i,j}^2$
   - ☐ $d_{i,j} = \frac{1}{c_{i,j}}$
   - ☐ $d_{i,j} = \alpha c_{i,j}$, donde $\alpha > 0$
   - ☐ $d_{i,j} = c_{i,j} + \alpha$, donde $\alpha > 0$

- ☐ $d_{i,j} = \alpha c_{i,j} + \beta$ donde $\alpha, \beta > 0$
- ☐ Ninguno de los anteriores

(d) (5 puntos) Selecciona todas las opciones en las cuales la selección de $d_{i,j}$ haga que la búsqueda UCS sea equivalente a la búsqueda *greedy* primero el mejor si consideramos a $h(n)$ como una heurística admisible evaluada en el nodo n (i.e. $h(i)$ es el valor de la heurística en el nodo i).

- ☐ $d_{i,j} = h(i) - h(j)$
- ☐ $d_{i,j} = h(j) - h(i)$
- ☐ $d_{i,j} = \alpha h(i)$, donde $\alpha > 0$
- ☐ $d_{i,j} = \alpha h(j)$, donde $\alpha > 0$
- ☐ $d_{i,j} = c_{i,j} + h(j) + h(i)$
- ☐ Ninguno de los anteriores

5. (20 puntos) Vamos a realizar un super exitante juego llamado *el Gato de* 2×2. Este juego es como el juego del gato, pero con solamente 4 casillas. Como cosa adicional, a los jugadores se les permite *pasar*. Las X siempre empiezan.

(a) (5 puntos) Dibuja el árbol de juego hasta una profundidad de 2. No agregues posiciones que sean rotaciones o reflexiones de la misma jugada. El árbol deberá tener al final 5 nodos hoja.

(b) (5 puntos) Si la función de utilidad es el número de X menos el número de O en el tablero, marca los valores de cada nodo, e indica cuales ramas serían podadas si utilizamos un algoritmo de poda α–β, y un ordenamiento de izquierda a derecha, de acuerdo a tu propio árbol generado.

(c) (5 puntos) Si quisiéramos resolver completamente el juego, explica porque en este caso la poda α–β con un ordenamiento de jugadas apropiado sería mucho mejor que el algoritmo básico de *minimax*.

(d) (5 puntos) Discute brevemente como debería de modificarse el algoritmo de *minimax* o de poda α–β para poder resolver el juego de *el Gato suicida de* 2×2, en el cual, gana el jugador que pierde. ¿Cual sería la mejor jugada del primer jugador?