Pregatire examen 27.01.2024

C1

1.1

Care este tipul de Inteligență Artificială mai potrivit pentru un sistem conceput să răspundă corect, dacă este posibil, la orice întrebare formulată în limbaj natural? 5 points

- [x] Comportament rațional.
- [ ] Gândire umană.
- [x] Comportament uman.
- [ ] Gândire rațională.

Tipul predominant: Comportament rațional (conteaza mai mult sa isi atinga scopul, decat sa se comporte uman.). Aditional, se poate considera si comportament uman.

"…să răspundă corect, dacă este posibil, la orice întrebare" -> Comportament rațional

"să răspundă ... la întrebare formulată în limbaj natural" -> Comportament uman

1.2

1. (1p) Propuneți un test ce permite identificarea unei Inteligențe Artificială ca fiind de tip 1 (se comportă ca un om).

Testul Turing

1.3

1. (1p) De ce trecerea testului Turing nu mai este suficientă pentru a demonstra Inteligența Artificială?

Weak AIs pot trece testul Turing. The Chinese Room Argument (Intelegerea nu este identica cu simularea intelegerii).

C2, C3

2.1

Care din următoarele reprezentări pentru o stare a problemei "Turnurile din Hanoi" (https://ro.wikipedia.org/wiki/Turnul_din_Hanoi), cu n tije și m piese, permit recuperarea unei soluții? 5 points

Check all that apply.

- [x] (n, m, p1, p2, ..., pm), unde n este numărul de tije, m numărul de piese și p1, p2, ..., pm este lista cu tijele pe care se află cele m piese
- [x] (n, p1, p2, ..., pm), unde n este numărul de tije, p1, p2, ..., pm este lista cu tijele pe care se află cele m piese
- [ ] (m, p1, p2, ..., pm), unde m este numărul de piese și p1, p2, ..., pm este lista cu tijele pe care se află cele m piese
- [x] (n, (p11,...,px1), (p12,...,py2), ..., (p1n,...,pzn)), unde n este numărul de tije și (p11,...,px1), (p12,...,py2), ..., (p1n,...,pzn) sunt liste cu piesele de pe fiecare tijă, în ordinea crescătoare a tijelor și a pieselor

O stare include TOATE informatiile necesare.

2.2

Care din următoarele euristici permit totdeauna recuperarea unei soluții, dacă ea există, folosind strategia Hillclimbing, pentru problema "Turnurile din Hanoi" (https://ro.wikipedia.org/wiki/Turnul_din_Hanoi), cu n tije și m piese?

Check all that apply.

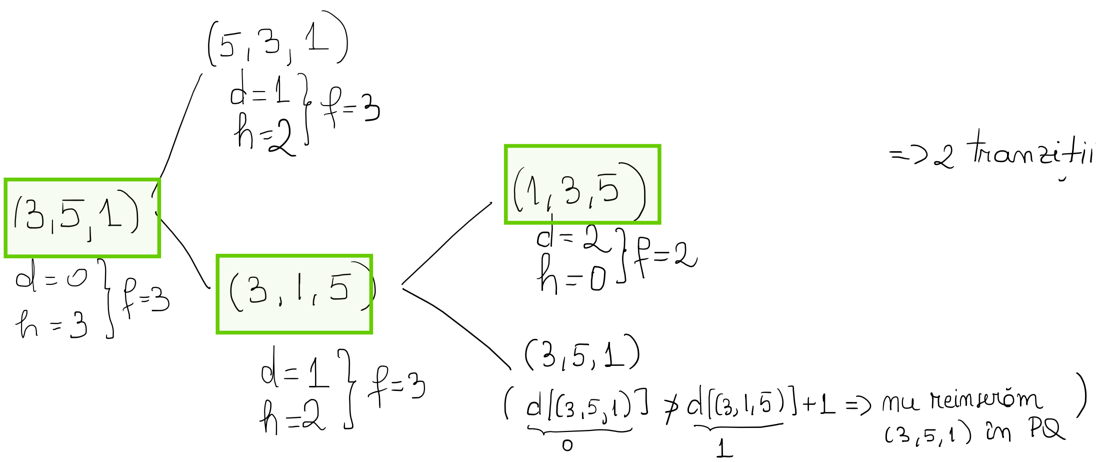
- [ ] m - tn, unde m este numărul de piese și tn este numărul de piese de pe tija destinație n
- [x] tn, unde tn este numărul de piese corect plasate pe tija destinație (așa cum sunt în starea finală)
- [ ] $suma_i$ (t_i * i), unde i ia valori de la 1 la n, t_i este numărul de piese de pe tija i
- [ ] t1, unde t1 este numărul de piese de pe tija inițială

Hill Climbing alege un vecin cu o valoare a euristicii cel putin la fel de buna ca cea a starii curente.

Pe un drum de la starea initiala la cea finala, numarul de piese de pe tija destinatie/tija initiala /suma ponderata a nr de pierse nu reprezinta o functie monotona.

2.3

Fie problema sortarii crescatoare a unei liste de n valori numerice cu restrictia ca nu putem schimba pozitia unui element din lista decat cu a unui element vecin. Folosind strategia A* si euristica h(s)=n- numarul de elemente plasate deja in pozitia corecta (cea din starea finala), cate tranzitii sunt necesare pentru a sorta crescator instanta (3,5,1)?



PQ: [((3,5,1), f=3)]

[~~((3,5,1), f=3)~~ ((3,1,5), f=3) ((5,3,1), f=3)]

[~~((3,1,5), f=3)~~ ((1,3,5), f=2) ((5,3,1), f=3)]

[~~((1,3,5), f=2)~~ ((5,3,1), f=3)]

C4

4.1

4. (1.5p) Considerăm următoarea problemă de satisfacere a restricțiilor: variabilele X, Y, Z cu domeniile: X: {1, 2, ...10}, Y: {5, 6, ...15}, Z: {5, 6, ...20} şi restrictiile X > Y, Y + Z = 12, X + Z = 16. Aplicați algoritmul de consistență arc pentru a actualiza domeniile variabilelor.

$$Q = [\underbrace{\cancel{(x,y)}, \cancel{(y,x)}, \cancel{(y,z)}, \cancel{(z,y)}, \cancel{(x,z)}, \cancel{(z,x)}}_{\text{puse inițial în Q}}, \cancel{(y,x)}, \cancel{(z,x)}, \cancel{(x,y)}, \cancel{(z,y)}, \cancel{(x,y)}, \cancel{(z,y)}, \cancel{(x,z)}, \cancel{(y,z)}, \cancel{(y,x)}, \cancel{(z,x)}, \cancel{(x,z)}, \cancel{(y,z)}, \cancel{(x,y)}, \cancel{(z,y)}$$

D	x	y	z
	~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~	5 6 ~~7~~ ~~8~~ ~~9~~ ~~10~~	~~5~~ 6 7 ~~8~~ ~~9~~ ~~10~~
	~~6~~ ~~7~~ ~~8~~ 9 10	~~11~~ ~~12~~ ~~13~~ ~~14~~ ~~15~~	~~11~~ ~~12~~ ~~13~~ ~~14~~ ~~15~~
			~~16~~ ~~17~~ ~~18~~ ~~19~~ ~~20~~

- (x,y) $x > y$
  pt. $x \in \{1,2,3,4,5\}$ $\nexists$ $y \in D(y)$ a.î. $x > y$
  $\Rightarrow$ ștergem $\{1,2,3,4,5\}$ din $D(x)$
  și inserăm $(y,x), (z,x)$ în Q

- (y,x) $y < x$
  pt. $y \in \{10,..,15\}$ $\nexists$ $x \in D(x)$ a.î. $y < x$ $\Rightarrow$ ștergem $\{10,..,15\}$ din $D(y)$
  și inserăm $(x,y), (z,y)$ în Q

- (y,z) $y+z=12$
  $D(y) = D(y) \setminus \{8,9\}$; inserare în Q: $(x,y), (z,y)$

- (z,y) $y+z=12$
  $D(z) = D(z) \setminus \{8,...,20\}$; inserare în Q: $(x,z), (y,z)$

- (x,z) $x+z=16$
  $D(x) = D(x) \setminus \{6,7,8\}$; inserare în Q: $(y,x), (z,x)$

- (z,x) $x+z=16$
  $D(z) = D(z) \setminus \{5\}$; inserare în Q: $(x,z) (y,z)$

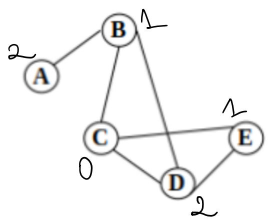
- (y,x) $y < x$, arcul $y \to x$ consistent
- (z,x) $x+z=16$, $z \to x$ consistent
- (x,y) $x > y$, $x \to y$ consistent
- (z,y) $y+z=12$, $z \to y$ consistent
- (x,z) $x+z=16$, $x \to z$ consistent

- (y,z) $y+z=12$
  $D(y) = D(y) \setminus \{7\}$; inserare în Q: $(x,y) (z,y)$
- $(y,x), (z,x), (x,z), (y,z), (x,y), (z,y)$ consistente

$$\Rightarrow \text{domenii finale} \begin{cases} D(x) = \{9, 10\} \\ D(y) = \{5,6\} \\ D(z) = \{6,7\} \end{cases}$$

4.2

5. (1.5p) Considerăm următorul graf de constrângeri asociat unei probleme de satisfacere a restricțiilor. Variabilele au următoarele domenii: A: {2}, B: {1,2}, C: {0, 1, 2}, D: {0, 1, 2}, E: {1, 2}. Aplicați algoritmul Backtracking + Forward checking + MRV (Minimum remaining values) pentru a identifica o soluție sau inconsistența.



| | | | | |

{A: 2, B: 1,2, C: 0,1,2, D: 0,1,2, E: 1,2}

↓

| A = 2 | | | | |

(MRV : var cu dom. minim = A)

{A:2, B: 1, ~~2~~, C: 0,1,2, D: 0,1,2, E: 1,2}

↓

| A = 2 | B = 1 | | | |

(MRV: var. cu dom. minim = B)

{A:2, B:1, C: 0, ~~1~~, 2, D: 0, ~~1~~, 2, E: 1,2}

↓

| A = 2 | B = 1 | C = 0 |

MRV → se va alege C, D sau E

{A:2, B:1, C:0, D: ~~0~~, 2, E: 1,2}

↓

| A = 2 | B = 1 | C = 0 | D = 2

MRV → se alege var D

{A:2, B:1, C:0, D:2, E: 1, ~~2~~}

↓

| A = 2 | B = 1 | C = 0 | D = 2 | E = 1 |

soluție

4.3

Care din următoarele afirmații referitoare la probleme de satisfacere a restricțiilor sunt adevărate? 5 points

- [ ] Algoritmul Min-conflicts determină întotdeauna o soluție pentru o problemă de satisfacere a restricțiilor
- [ ] Dacă graful de constrângeri asociat unei probleme de satisfacere a restricțiilor binară are structură arborescentă, atunci problema poate fi rezolvata în timp liniar în numărul de resticții.
- [x] Putem combina euristica Minimum-remaining-values cu algoritmul Min-conflicts
- [ ] Euristica Minimum-remaining-values este utilizată pentru a identifica valoarea variabilei de asignat
- [x] O asignare consistentă este o asignare care satisface restricțiile

Min Conflicts este o euristica utilizata pentru cautare locala.

Algoritmul pentru rezolvarea problemelor de satisfacere a restrictiilor pt grafuri de constrangere de tip arbori are complexitatea O(n*d^2) , n= numar variabile, d= cardinal maxim domeniu var.

In loc sa selectam variabilele random in MinConflicts, putem utiliza MRV ca sa alegem variabilele (din setul celor care sunt incluse in constrangeri curent incalcate).

Cu MRV selectam variabila cu domeniu minim.

C5

5.1

Care din următoarele afirmații sunt adevărate pentru jocul de șah?

- [x] Există un echilibru Nash.
- [ ] Este un joc zero-sum și simultan. alternativ
- [x] Este un joc cu informație perfectă.
- [x] Este un joc simetric.

Nash's Theorem (Nash, 1950). Any game with a finite number of players and a finite number of actions has a mixed-strategy Nash equilibrium.

5.2 Optimalitate Pareto, strategii dominante, echilibre Nash pure

I

P_1 \ P_2	L	R
L	1,1	0,0
R	0,0	1,1

Ech. Nash pure: (L,L), (R,R)

Rezultate optime Pareto: (L,L), (R,R)

Nu ∃ strategii dominante pure

II

P_1 \ P_2	B	F
B	2,1	0,0
F	0,0	1,2

Ech Nash pure: (B,B), (F,F)

Rezultate optime Pareto: (B,B), (F,F)

Nu ∃ strategii dominante pure

III

		P_2: H	T
P_1	H	1, -1	-1, 1
	T	-1, 1	1, -1

Nu ∃ echilibre Nash pure
Rezultate optime Paretto: (H,H), (H,T), (T,H), (T,T)
Nu ∃ strategii dominante pure

IV

		P_2: A	B
P_1	A	-1, -1	-4, 0
	B	0, -4	-3, -3

Ech. Nash pure: (B,B)
Rez. optime Pareto: (A,A), (A,B), (B,A)
P_1 are strategia pură dominantă B
P_2 are strategia pură dominantă B

Chiar dc. ∃ echilibru Nash pur, poate ∃ și echilibru Nash mixt.

ex:

I

		P_2: L q	R $1-q$
P_1	L p	1, 1	0, 0
	R $1-p$	0, 0	1, 1

$$\underbrace{\text{payoff dc. } P_1 \text{ alege } L}_{1\cdot q + 0\cdot(1-q)} = \underbrace{\text{payoff dc. } P_1 \text{ alege } R}_{0\cdot q + 1\cdot(1-q)}$$

$$\underbrace{\text{payoff dc. } P_2 \text{ alege } L}_{1\cdot p + 0\cdot(1-p)} = \underbrace{\text{payoff dc. } P_2 \text{ alege } R}_{0\cdot p + 1\cdot(1-p)}$$

$\Rightarrow p = 1/2,\ q = 1/2$

⇒ Ech. Nash mixt e format din strategiile { pt. P_1: alege L cu probab 1/2 și R cu probab 1/2; pt. P_2: alege L cu probab 1/2 și R cu probab 1/2 }

5.3

6. (1.5p) Fie jocul din figura de mai jos. Ce strategie este dominată pentru Rose? Care este echilibrul Nash (pur sau mixt) al jocului? Cât câștigă Rose și Colin în situația de echilibru?

Dc Colin { alege D ⇒ Rose câștigă $-1\cdot p + 3(1-p)$; alege E ⇒ Rose câștigă $2\cdot p - 3(1-p)$ }

Colin alege D cu probab q și E cu prob $1-q$

⇒ Rose câștigă la echilibru:

$q\cdot(-1\cdot p + 3(1-p)) + (1-q)(2p - 3(1-p)) = 0.333$ (înlocuim $p=1/2, q=5/9$)

		Colin: D q	E $1-q$
Rose	A	~~-2, 3~~	~~-4, 0~~
	B p	-1, 2	2, -1
	C $1-p$	3, -2	-3, 1

Startegia dominate pentru Rose: (-2,-4). O eliminam pentru a putea determina echilibrul / echilibrele Nash.

Nu exista echilibre Nash pure.

Cautam echilibru Nash mixt: (Rose alege B cu probab p și C cu probab $1-p$, Colin alege D cu probab q și E cu probab $1-q$)

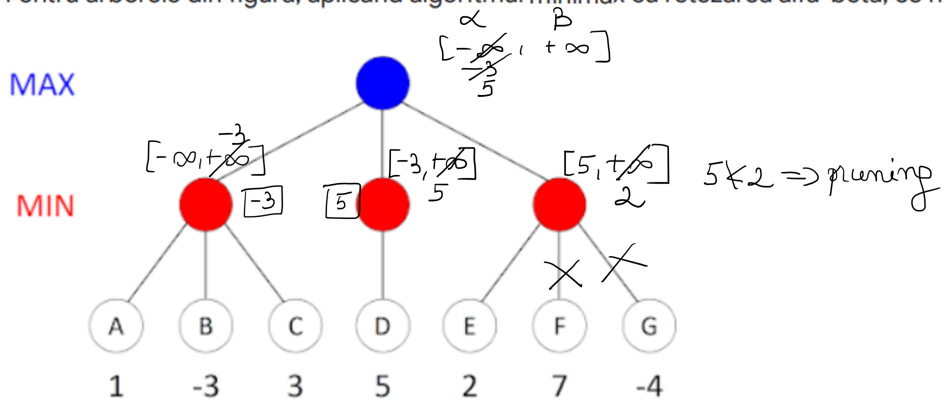
payoffs pt. Rose: $$\underbrace{\text{payoff dc. Rose alege B}}_{-1\cdot q + 2(1-q)} = \underbrace{\text{payoff dc. Rose alege C}}_{3\cdot q + -3(1-q)}$$

payoffs pt. Colin: $$\underbrace{\text{payoff dc. Colin alege D}}_{p\cdot 2 - 2(1-p)} = \underbrace{\text{payoff dc. Colin alege E}}_{-1\cdot p + 1\cdot(1-p)}$$

$\Rightarrow q = \frac{5}{9}$, $p = \frac{1}{2}$

5.4

Pentru arborele din figură, aplicând algoritmul minimax cu retezarea alfa-beta, ce noduri nu vor mai trebui evaluate? 5 p



Check all that apply.

- [ ] Nodurile B și G
- [ ] Nodurile E, F și G
- [ ] Nodurile F și G
- [ ] Nodurile D, E, F și G
- [ ] Arborele nu este corect construit deoarece factorul de ramificare de pe nivelul MIN nu este egal pentru toate nodurile; prin urmare, algoritmul nu se poate aplica
- [ ] Nodul G

C6

6.1

Considerăm datele de mai jos. Aplicați regula de antrenare a perceptronului pentru a actualiza ponderile. Considerați funcția de activare treaptă, pragul egal cu 0.2 și rata de învățare de 0.1. Ponderile inițiale sunt egale cu 0.3 și -0.1. Care din următoarele afirmații sunt adevărate?

Inputs x_1	Inputs x_2	Desired output $Y_d = t$
0	0	0
0	1	0
1	0	0
1	1	1

$$\begin{cases} o = f(\Sigma w_i x_i + b) \\ \Delta w_i = \eta (t - o) x_i \\ \Delta b = \eta (t - o) \\ w \mathrel{+}= \Delta w_i \\ b \mathrel{+}= \Delta b \end{cases}$$

$w_1 = 0.3$ $w_2 = -0.1$ $b = 0$ $\eta = 0.1$ $f(x) = \begin{cases} 1, & x \geq 0.2 \\ 0, & \text{altfel} \end{cases}$

- $o = f(0 \cdot 0.3 - 0.1 \cdot 0 + 0) = f(0) = 0$, $t = 0$
  $\Downarrow$
  $\Delta w_1 = \Delta w_2 = \Delta b = 0$, w_1, w_2 și b rămân neschimbate
- $o = f(0 \cdot 0.3 - 0.1 \cdot 1 + 0) = f(-0.1) = 0$, $t = 0$
  $\Downarrow$
  $\Delta w_1 = \Delta w_2 = \Delta b = 0$, w_1, w_2 și b rămân neschimbate
- $o = f(1 \cdot 0.3 - 0.1 \cdot 0 + 0) = f(0.3) = 1$, $t = 0$
  $\Downarrow$
  $\Delta w_1 = 0.1 \cdot (0 - 1) \cdot 1 = -0.1$, $\Delta w_2 = 0.1(0 - 1) \cdot 0 = 0$, $\Delta b = 0.1(0 - 1) = -0.1$
  $\Rightarrow w_1 = 0.2$, $w_2 = -0.1$, $b = -0.1$

• $o = f(1 \cdot 0.2 - 0.1 \cdot 1 - 0.1) = f(0) = 0$, $t = 1$

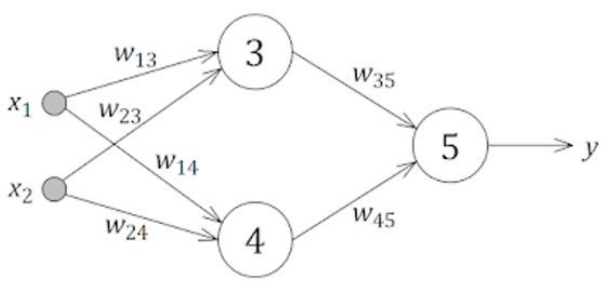
$\Downarrow$

$$\left.\begin{aligned} \Delta w_1 &= 0.1(1-0) \cdot 1 = 0.1 \\ \Delta w_2 &= 0.1(1-0) \cdot 1 = 0.1 \\ \Delta b &= 0.1(1-0) = 0.1 \end{aligned}\right| \Rightarrow \boxed{\begin{aligned} w_1 &= 0.3 \\ w_2 &= 0 \\ b &= 0 \end{aligned}}$$

În versiunea de mai sus am inclus și biasul. De obicei, când nu se menționează nimic de acesta, biasul poate fi omis.

6.2

✓ Fie rețeaua neuronală din figură. Valorile ponderilor sunt următoarele: w13 = 1, w23 = –1, w14 = 0, w24 = 1, w35 = –1, w45 = 4. Pragurile neuronilor sunt toate egale cu 0. Toți neuronii au funcția de activare ReLU: f(x) = max(0, x). Care este ieșirea rețelei pentru intrările x1 = -1 și x2 = -2? **5/5**



$o = f(\Sigma x_i w_i + b)$, $b = 0$ (din enunț)

$o_3 = f(x_1 \cdot w_{13} + x_2 \cdot w_{23}) = f(-1 \cdot 1 + (-2) \cdot (-1)) = f(1) = 1$

$o_4 = f(x_1 \cdot w_{14} + x_2 \cdot w_{24}) = f(-1 \cdot 0 + (-2) \cdot 1) = f(-2) = 0$

$o_5 = f(o_3 \cdot w_{35} + o_4 \cdot w_{45}) = f(1 \cdot -1 + 0 \cdot 4) = f(-1) = \boxed{0}$

6.3

Care din afirmațiile de mai jos sunt adevărate? 5 points

- [ ] Putem utiliza regula de antrenare a perceptronului pentru a clasifica orice problemă de clasificare binară.
- [ ] Regula de antrenare a perceptronului converge și atunci când clasele nu sunt liniar separabile.
- [x] Algoritmul backpropagation poate converge într-un minim local al funcției de eroare.

Un Perceptron poate reprezenta doar functii binare separabile liniar.

Da, Backpropagation foloseste Gradient Descent, care face o cautare locala.