

NUME, GRUPĂ: _____
DATA: _____

**Examen la disciplina Inteligență Artificială,
partea de "Căutarea și Reprezentarea Cunoștințelor"
- Varianta 1 -**

1. Se consideră problema blocurilor (cu S stive și NB blocuri; S și NB naturale, $S \geq 3$, $NB \geq 4$, NB par) cu următoarele modificări:

- $NB-1$ blocuri conțin o informație numerică de forma 0 sau 1.
- Există un bloc special, numit blocul "not", peste care nu poate exista niciodată un alt element și care nu se poate găsi niciodată pe stivele din capetele configurației
- Ca la problema blocurilor clasică, putem **muta** un bloc dintr-un vârf de stivă pe alt vârf de stivă, cu următoarele restricții:
 - Blocul "not" se poate muta pe orice stivă mai puțin cele din capete (prima și ultima).
 - Când blocul "not" este mutat pe o stivă nevidă, valoarea tuturor blocurilor de pe acea stivă se modifică din 0 în 1 și din 1 în 0.
 - Peste blocul "not" nu putem muta niciun bloc numeric.
- **Costul** mutării unui *bloc numeric* este egal cu numărul de pe bloc + 1 (blocurile 0 au costul 1, blocurile 1 au costul 2). Costul mutării blocului "not" este $1 + nr_schimbari$, unde $nr_schimbari$ e numărul de blocuri numerice afectate de mutarea sa.
- **Scopul** este să ajungem la o configurație în care există exact o stivă doar cu blocuri 0, exact o stivă doar cu blocuri 1, iar ambele stive conțin exact $NB/2$ blocuri (NB precizat la început).

Notăm cu SV numărul de stive vide.

Care dintre următoarele afirmații sunt adevărate?

- a. Pentru $S \geq 4$, nu există nicio stare inițială pentru care să nu existe soluție.
- b. Numărul de succesori valizi ai unei stări oarecare este $2 \cdot (S - SV - 2)$
- c. Presupunând că blocul "not" se află singur pe o stivă (sub el nu mai e alt bloc). Numărul de succesori valizi ai unei stări oarecare este $(S - SV - 1) \cdot (S - 2) + S - 3$
- d. Presupunând că blocul "not" se află singur pe o stivă (sub el nu mai e alt bloc). Numărul de succesori valizi ai unei stări oarecare este $(S - SV - 2) \cdot (S - 1) + S - 3$
- e. Presupunând că blocul "not" se află singur pe o stivă (sub el nu mai e alt bloc). Numărul de succesori valizi ai unei stări oarecare este $(S - 1) + (S - SV) \cdot (S - 1)$
- f. Pentru $S = 3$ nu putem avea o stare inițială conținând în configurație o stivă vidă.
- g. niciuna dintre opțiunile de mai sus nu e corectă.

2. Considerăm enunțul problemei anterioare. Care este o estimatie admisibilă pentru o stare?

- a. $S - SV - 3$, unde SV e numărul de stive vide din starea respectivă.
- b. SV , unde SV e numărul de stive vide din starea respectivă.
- c. $|Nr(0) - Nr(1)|$, unde $Nr(x)$ e numărul de blocuri cu valoarea x .
- d. 0 indiferent de stare.
- e. 1 dacă starea nu e scop și 0 dacă e stare scop.

- f. 1 indiferent de stare.
- g. niciuna dintre opțiunile de mai sus nu e corectă.

3. Considerăm un arbore minimax de adâncime maximă A și N noduri (inclusiv rădăcina) generat pentru un joc oarecare. Care dintre următoarele afirmații sunt adevărate?

- a. Un nod MIN nu poate fi frate cu un nod MAX
- b. Dacă aplicăm alpha-beta pe aceeași stare rădăcină putem să evităm calcularea a maxim $N/2$ noduri
- c. Dacă aplicăm alpha-beta pe aceeași stare rădăcină putem să evităm calcularea a maxim $N-1$ noduri
- d. Dacă aplicăm alpha-beta pe aceeași stare rădăcină putem întotdeauna să evităm calcularea a minim unui nod
- e. Alpha-beta nu va elimina (reteza) niciodată primul fiu al unui nod.
- f. Frunzele sunt întotdeauna noduri MIN.
- g. niciuna dintre opțiunile de mai sus nu e corectă.

4. Pentru N număr natural ($5 \leq N \leq 10$) considerăm o tablă de joc de dimensiune $N \times N$ în care sunt NL lăzi notate pe hartă cu câte un număr G reprezentând greutatea fiecărei lăzi. NL și G sunt numere naturale nenule. Pe hartă avem și un omuleț notat cu X , așa cum se vede în desen (desenul arată doar un exemplu de stare inițială; exercițiul nu se referă doar la acest caz specific de stare inițială).

	3			10
				8
	5			
X	14			5

Scopul omulețului este să împingă un număr K de lăzi (K natural nenul, $K \leq NL$) în colțul din stânga sus al matricei (poziția 0,0). Omulețul poate împinge o ladă doar cu câte o poziție și doar într-un spațiu liber. În plus, el trebuie să fie în spațiul vecin lăzii pe aceeași direcție dar în sens opus față de sensul mutării (de exemplu, pentru a muta o ladă în dreapta, omulețul trebuie să fie în stânga ei; cum e pe desen în cazul lăzii 14). În starea inițială pe poziția (0,0) nu poate exista nicio ladă și aceasta nu poate reprezenta nici poziția inițială a omulețului. O ladă ajunsă (împinsă) în locația (0,0) dispare. Omulețul se poate deplasa și poate muta lăzi doar cu câte o poziție în direcțiile: sus, jos, stânga și dreapta (nu și pe diagonală). Fiecare deplasare **costă**:

- 1 dacă omulețul nu împinge o ladă
- greutatea lăzii dacă omulețul împinge o ladă

Notății:

- Coordonatele omulețului: linie(om), coloana(om).
- Coordonatele unei lăzi: linie(lada[i]) și coloana(lada[i]).
- Distanța Manhattan: $\text{distMH}(\text{linie1}, \text{coloana1}, \text{linie2}, \text{coloana2}) = |\text{linie1} - \text{linie2}| + |\text{coloana1} - \text{coloana2}|$
- Greutatea lăzii x: greutate(x)

Care dintre estimățiile de mai jos sunt admisibile pentru această problemă?

- Dacă omulețul a scos deja K1 lăzi din hartă în starea curentă, o estimatie admisibilă este $K - K1$
- O estimatie admisibilă este $\text{distMH}(\text{linie}(\text{om}), \text{coloana}(\text{om}), 0, 0)$ indiferent de câte lăzi au fost scoase de pe hartă
- Pentru $NL \geq 4$ există o configurație inițială (în spațiu stărilor posibile) fără soluție.
- Pentru $NL = 1$ problema întotdeauna are soluție.
- O estimatie admisibilă pentru o stare în care omul mai are de scos un număr nenul de lăzi, este $\text{distMH}(\text{linie}(\text{om}), \text{coloana}(\text{om}), \text{linie}(\text{lada_min}), \text{coloana}(\text{lada_min})) * \text{greutate}(\text{lada_min})$, unde lada_min e lada cu cea mai mică greutate din configurație
- niciuna dintre opțiunile de mai sus nu e corectă.

5. Care dintre următoarele afirmații sunt adevărate pentru problema X și 0? (enunț în anexă)

- Pentru starea reprezentată mai jos, calculând arborele minimax în totalitate, fără a impune o adâncime maximă, numărul de noduri din arbore ar fi egal cu $24=4!$

x	0	
	x	
x	0	

- Pentru starea reprezentată mai jos, calculând arborele minimax în totalitate, fără a impune o adâncime maximă, adâncimea arborelui este 1

x	0	
	x	
x	0	

- O stare finală a jocului are ca proprietate că fie a câștigat X, fie a câștigat 0.
- Arborele minimax pentru X și 0 întotdeauna va avea cel mult 10 niveluri.
- Pentru starea reprezentată mai jos, calculând arborele minimax în totalitate, fără a impune o adâncime maximă, numărul de noduri din arbore ar fi egal cu 5

x	0	x
0	x	
	0	x

- Un nod frunză din arborele Minimax, pentru jocul x și 0 conține întotdeauna o tablă fără locuri libere (tabla e în întregime completată).

g. niciuna dintre opțiunile de mai sus nu e corectă.

6. Se dă o problemă de căutare cu costuri strict pozitive pe tranziții. Se dau o estimatie admisibilă $\hat{h}_1(\text{nod})$ și o estimatie neadmisibilă $\hat{h}_2(\text{nod})$, oarecare. Care dintre formulele următoare ar obține din \hat{h}_1 și \hat{h}_2 o nouă estimatie $\hat{h}(\text{nod})$ cu siguranță **neadmisibilă**?

Observație: Formula trebuie să fie adevărată pentru orice \hat{h}_1 , orice \hat{h}_2 și orice nod din graf, nu doar pentru cazuri particulare. Știm că $\hat{h}_1(\text{nod}) \geq 0$ și $\hat{h}_2(\text{nod}) \geq 0$ pentru orice nod din graf.

- a. $\hat{h}(\text{nod}) = \hat{h}_2(\text{nod}) + \hat{h}_1(\text{nod}) / 3$
- b. $\hat{h}(\text{nod}) = \hat{h}_1(\text{nod}) + \hat{h}_2(\text{nod}) / 3$
- c. $\hat{h}(\text{nod}) = \hat{h}_2(\text{nod}) - \hat{h}_1(\text{nod})$
- d. $\hat{h}(\text{nod}) = 2 * \hat{h}_2(\text{nod}) - \hat{h}_1(\text{nod})$
- e. $\hat{h}(\text{nod}) = \hat{h}_1(\text{nod}) * \hat{h}_2(\text{nod})$
- f. niciuna dintre formulele de mai sus nu e corectă

7. Considerăm următorul joc:

Pentru N număr natural ($5 \leq N \leq 10$) considerăm o tablă de joc de dimensiune $N \times N$ și 2 jucători (X și O).

Jucătorului X îi corespunde prima linie, iar lui O ultima. Jucătorul X mută primul.

O mutare constă în una dintre următoarele acțiuni:

- 1. Plasarea unui simbol nou pe linia proprie.
- 2. Mutarea unui simbol cu o singură poziție pe orizontală (în stânga sau dreapta) sau pe verticală, înspre linia adversarului (în jos pentru x și în sus pentru O).

Restricții de mutare:

- 1. Nu pot exista mai mult de 2 simboluri ale aceluiași jucător pe vreo linie
- 2. Plasarea unui simbol nou se poate realiza doar dacă pe coloana respectivă nu mai sunt simboluri proprii.

Jucătorul j poate captura o piesă a adversarului, ja , dacă piesa adversarului se află între două piese ale jucătorului j aflate pe aceeași linie cu aceasta, iar cele două piese ale jucătorului j au între ele cel mult $N/3$ spații.

Un jucător **câștigă** fie când a capturat un număr $K_MAX > 3$ de simboluri, fie când a ajuns cu un simbol pe linia adversarului.

Care dintre variantele de mai jos oferă o funcție de evaluare care să arate în mod corect cât de favorabilă este starea jocului pentru calculator (MAX) (să aibă o valoare mai mare pentru stări mai favorabile și mai mică pentru stări mai nefavorabile) ?

- a. Numărul de simboluri ale lui MAX de pe linii pare din care scădem numărul de simboluri ale lui MIN de pe coloane impare.
- b. Câte configurații de 3 simboluri pe poziții consecutive (pe linie/coloană/diagonală) are MAX din care scădem câte configurații de 3 simboluri are MIN
- c. Numărul de simboluri capturate ale lui MAX din care scădem numărul de simboluri capturate ale lui MIN .

- d. Considerăm LMIN linia celui mai depărtat simbol al lui MIN de linia sa de start și START_MIN, linia de start a lui MIN. Considerăm LMAX linia celui mai depărtat simbol al lui MAX de linia sa de start și START_MAX, linia de start a lui MAX. Atunci funcția de evaluare ar fi $|LMAX - START_MAX| - |LMIN - START_MIN|$
- e. Numărul de simboluri ale lui MAX la care adunăm numărul de simboluri ale lui MIN.
- f. Numărul de simboluri capturate ale lui MAX înmulțite numărul de simboluri capturate ale lui MIN.
- g. Numărul de locuri libere de pe tablă.

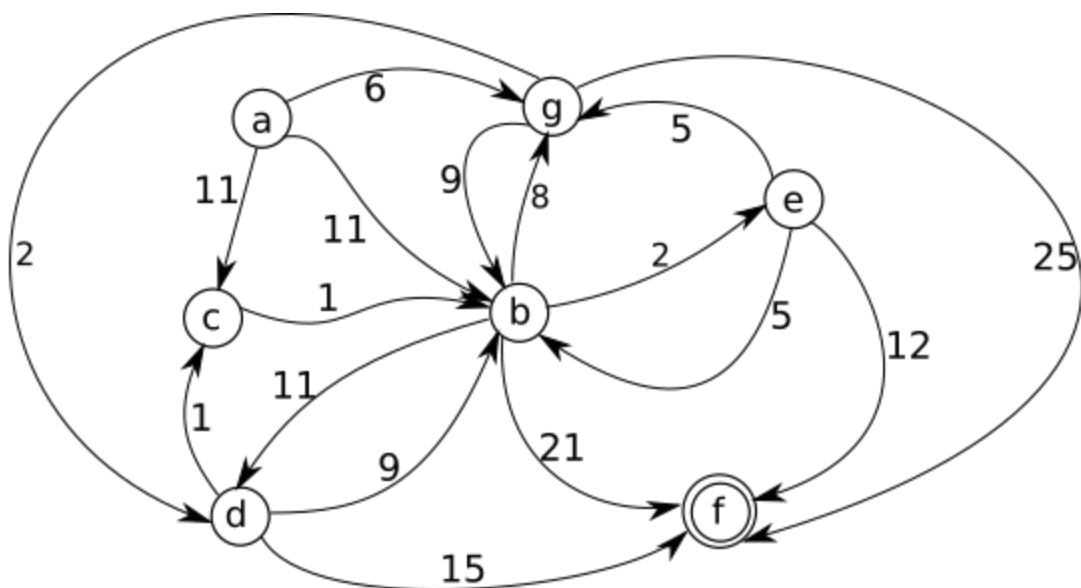
8. Pentru graful din imagine avem stările cozii OPEN **ordonate** pentru algoritmul A* de mai jos. În fiecare stare sunt precizate doar informațiile nodurilor din coadă. Considerăm nodul **start a**, iar nodul **scop f**. Prima stare e cea inițială a cozii OPEN (cu nodul start inclus). Toate celelalte stări sunt afișate în urma repetării următoarelor acțiuni:

- 1) Extrage nodul cu f minim (nu se afișează coada pentru această stare intermediară)
- 2) Adaugă succesorii în coadă astfel încât coada să fie mereu ordonată crescător după f (după această acțiune se afișează coada).

Observație: Pentru valori f egale, elementele sunt în ordinea descrescătoare a valorii g . În cazul în care f și g sunt egale, nodurile se ordonează după informație. În cazul în care un succesor are deja informația în CLOSED, dar găsim un cost mai bun pentru aceeași informație, îl mutăm înapoi în OPEN pentru expandare.

Stările cozii OPEN (numărul din stânga e numărul de ordine al stării):

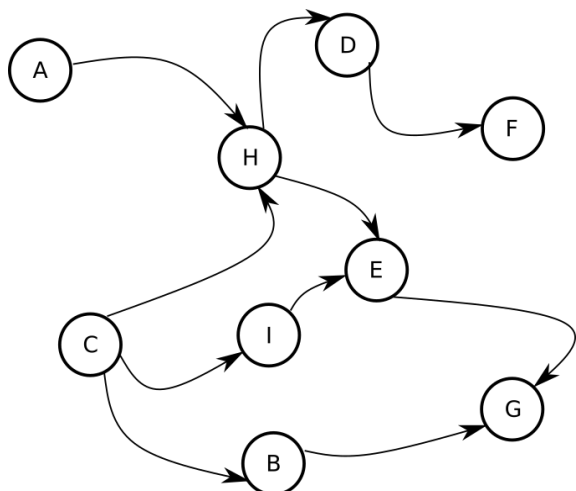
- 0: [a]
- 1: [c, b, g]
- 2: [b, g]
- 3: [g, e, f, d]
- 4: [d, e, f]
- 5: [c, f, e]
- 6: [b, f, e]
- 7: [f, e]



Care dintre următoarele afirmații sunt adevărate pentru **graful dat și stările cozii OPEN de mai sus**?

- Estimația (cu valorile scrise între paranteze drepte pentru fiecare nod) $a[10] \ b[1] \ c[3] \ d[5] \ e[4] \ f[0] \ g[6]$ este admisibilă și în concordanță cu coada OPEN.
- Estimația (cu valorile scrise între paranteze drepte pentru fiecare nod) $a[0] \ b[0] \ c[0] \ d[0] \ e[0] \ f[0] \ g[0]$ este admisibilă și în concordanță cu coada OPEN.
- Estimația (cu valorile scrise între paranteze drepte pentru fiecare nod) $a[1] \ b[1] \ c[1] \ d[1] \ e[1] \ f[1] \ g[1]$ este admisibilă și în concordanță cu coada OPEN.
- Estimația (cu valorile scrise între paranteze drepte pentru fiecare nod) $a[2] \ b[1] \ c[7] \ d[3] \ e[1] \ f[0] \ g[1]$ este admisibilă și în concordanță cu coada OPEN.
- Niciuna dintre opțiunile de mai sus nu e corectă.

9. Se dă următoarea topologie de rețea Bayesiană:

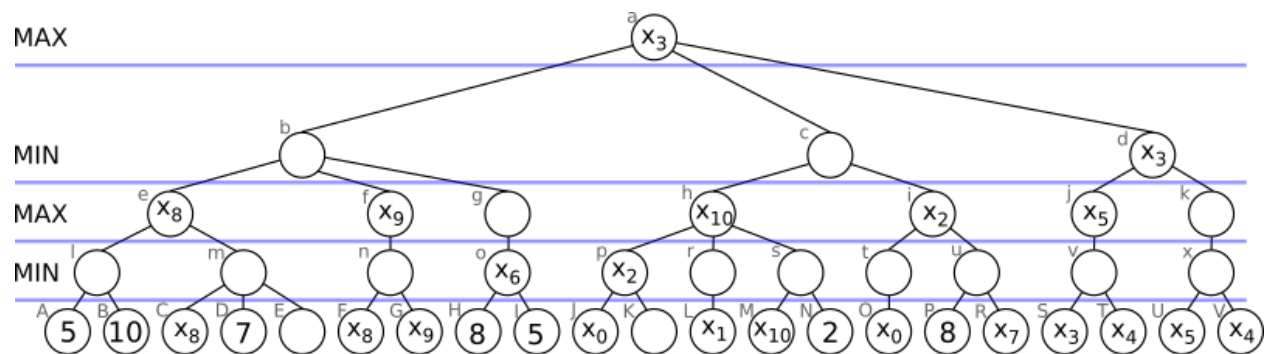


Care dintre următoarele afirmații sunt adevărate?

- Dacă am adăuga arcul $G \rightarrow C$ graful nu ar mai fi o topologie corectă pentru o rețea Bayesiană. Mulțimea $\{I, B, E\}$ d-separă mulțimea $\{C\}$ de mulțimea $\{E, G\}$.
- Graful dat nu este o topologie corectă pentru o rețea Bayesiană.
- Dacă am adăuga arcul $C \rightarrow A$ graful nu ar mai fi o topologie corectă pentru o rețea Bayesiană.
- Drumul de la nodul A la nodul F e blocat condiționat de mulțimea $\{D\}$
- niciuna dintre opțiunile de mai sus nu e corectă.

10. Se consideră arborele minimax din imagine, cu adâncime maximă 4 (rădăcina fiind considerată la adâncime 0). Se presupune că arborele a fost deja generat prin minimax, iar unele valori minimax au fost, apoi, șterse din imagine (nodurile fără conținut), fie înlocuite cu identificatori x_i (două noduri cu același identificator x_i au valorile minimax egale, însă putem avea $x_i = x_j$ pentru i diferit de j). Care dintre afirmațiile de mai jos sunt sigur adevărate având în vedere informația dată despre arbore?

Observație: frunzele sunt notate cu litere mari iar nodurile interne cu litere mici.



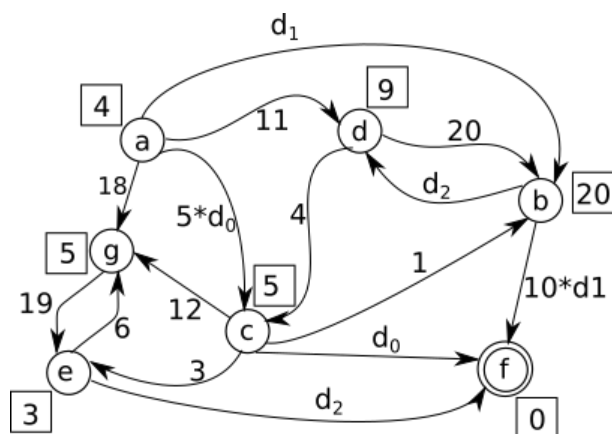
- x_2 este sigur egal cu x_0
- x_8 aparține sigur intervalului $[5, 7]$
- nodul b are sigur valoarea x_9
- x_3 este cea mai mare valoare dintre toate valorile nodurilor arborelui
- nodul c (literă mică) va avea sigur valoarea x_2
- x_7 este sigur mai mic strict decât x_2
- niciuna dintre opțiunile de mai sus nu e corectă.

11. Se dă graful orientat cu arce ponderate din imagine. Pentru unele arce ponderile nu sunt precizate, fiind înlocuite de identificatorii d_i . Euristică dată este una admisibilă (estimația pentru fiecare nod e trecută în pătrățelul de lângă nod).

Costurile arcelor sunt numere naturale nenule.

Nodul start este a și nodul f este nod scop.

Drumul de cost minim returnat de A* pentru nodul de start a este $a \rightarrow b \rightarrow d \rightarrow c \rightarrow f$ cu costul 14 și este unicul drum de acest cost.



Care dintre afirmațiile de mai jos sunt adevărate:

- d_0 are valoarea 5
- d_2 este întotdeauna mai mic strict decât 2
- $d_1 + d_2 = 5$ întotdeauna
- d_1 este 4 și d_2 este mai mare sau egal cu 1
- d_1 este mereu mai mic sau egal cu 1
- există în graf un drum cu costul 15
- niciuna dintre variantele de mai sus nu este corectă

12. Care dintre următoarele afirmații sunt adevărate pentru rețele Bayesiene (și topologiile lor)?

- Orice graf orientat cu arce având asociate ponderi numere reale poate reprezenta topologia unei rețele Bayesiene.
- O proprietate a topologiei unei rețele Bayesiene este faptul că este un graf orientat (direcționat).
- Întotdeauna gradul interior (numărul de arce care intră în nod) al unui nod este mai mic sau egal cu 1.
- Întotdeauna gradul exterior al oricărui nod (numărul de arce care ies din nod) este mai mare sau egal cu gradul său interior (numărul de arce care intră în nod).
- O proprietate a topologiei unei rețele Bayesiene este faptul că este un graf fără circuite (aciclic).
- niciuna dintre opțiunile de mai sus nu e corectă.