Opgaven Kunstmatige Intelligentie — 1 maart 2024

Opgave 1.

a. Denkt een schaakprogramma?
b. Denkt een (Nederlands-Engels) vertaalprogramma?
c. Denkt een C++-compiler?
d. Denkt Watson, the IBM-computer die in februari 2011 "Jeopardy!" won? En voldoet Watson aan de Turing-test? En ChatGPT?
e. Kan een computer denken?

Opgave 2.

a. De omgeving voor een agent kan statisch of dynamisch zijn. Wat betekent het tussengeval *semi-dynamisch*? Geef een voorbeeld.
b. De omgeving voor een agent kan deterministisch of stochastisch zijn. Hoe zou je een situatie inschatten waarbij een tegenstander zetten doet die je zelf niet kunt voorspellen? Geef van alle drie een voorbeeld.
c. Is de omgeving bij het spelen van een spelletje poker 1. ja/nee volledig observeerbaar, 2. ja/nee episodisch, 3. ja/nee deterministisch/strategisch? Geef duidelijke verklaringen van de antwoorden. En wat is het antwoord bij 2. op een pokertoernooi?

Opgave 3.

a. Geef een korte "PEAS" beschrijving van een meerpersoons autoracespel dat via internet gespeeld wordt, en waarbij met op afstand bestuurbare auto's gereden wordt. Vertel ook waar de "PEAS" afkortingen van zijn.
b. Maak een korte "PEAS" beschrijving voor de taak-omgeving van een ploeg robots die voetbal speelt. Geef hiertoe per letter P-E-A-S een tweetal steekwoorden.
c. En idem voor een student die een schriftelijk tentamen maakt.

Opgave 4.

Een agent bevindt zich in de linker of in de rechter kamer. De enige mogelijke acties zijn $\mathcal{L}$: ga naar de linker kamer en $\mathcal{R}$: ga naar de rechter kamer. Geef de toestand-actie-ruimte met de drie bereikbare *belief states* voor het sensorloze (conformante) geval.

Opgave 5.

Probeer een eenvoudig schaak-programma (witte dame en koning tegen zwarte koning) onder te brengen in elk van de volgende categorieën, door het steeds — in gedachten — iets krachtiger te maken: 1) reflex-gebaseerde agent, 2) idem, met toestand, 3) doel-gebaseerde agent, 4) nut-gebaseerde agent, 5) lerende agent.

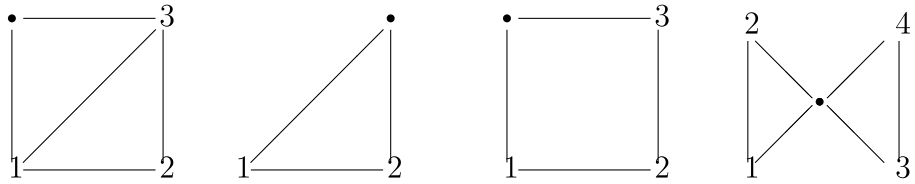
Opgave 6.

Probeer het programma van de tweede programmeeropgave onder te brengen in één of meer van de volgende categorieën: 1) reflex-gebaseerde agent, 2) idem, met toestand, 3) doel-gebaseerde agent, 4) nut-gebaseerde agent, 5) lerende agent.

Opgave 7.

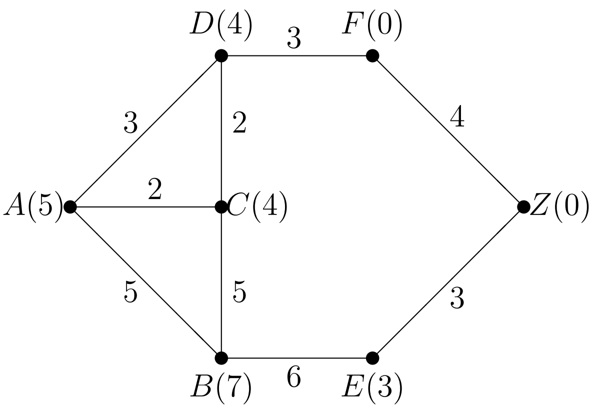
We bekijken *schuifpuzzels*, zoals de 15-puzzel. Bij zo'n schuifpuzzel heb je een graaf waarbij alle knopen behalve één een uniek nummer hebben. Je mag een getal naar zijn buur

schuiven als die buur geen nummer heeft. (De knopen blijven op hun plek zitten.) De vraag is: welke posities kun je vanuit een vaste bereiken?
a. Los de twee meest linkse onderstaande puzzels op. Oftewel: hoeveel verschillende patronen, die niet (door herhaald te schuiven) in elkaar over te voeren zijn, heb je? Geef eerst het aantal verschillende mogelijke posities.



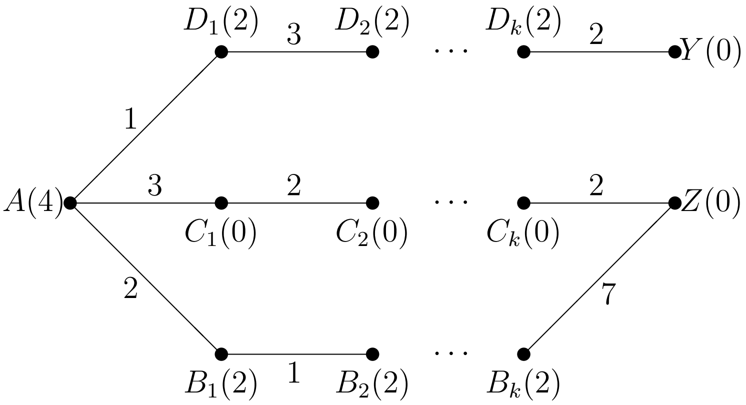
b. Hoeveel verschillende patronen, die niet in elkaar over te voeren zijn, heb je als de getallen $1, 2, \ldots, n-1$ op een "cirkel" liggen, waarbij ieder getal alleen met de eerst kleinere en eerst grotere verbonden is? (n is de lege plek, verbonden met 0 en $n-1$.)
Als voorbeeld (het tweede plaatje is al het geval $n = 3$) laat het derde plaatje de situatie met $n = 4$ zien (en dat is analoog aan de 15-puzzel eigenlijk de 3-puzzel).
c. En voor een "vlinderdasje" $\bowtie$ (zie het vierde plaatje)?
d. Hoe zou je een computerprogramma schrijven dat dit uitrekent?

Opgave 8. (opgave van tentamen 1 juni 2001)
a. Leg het A*-algoritme uit.
b. Wanneer heet een heuristiek *toelaatbaar* (= admissibel)?
c. Wat is de *pathmax equation* en waarvoor wordt deze gebruikt?
d. Bepaal een kortste pad van begin naar doel voor onderstaande graaf. Beginknoop is A, doelknoop is Z. De kostenfunctie staat naast de takken van de graaf.
Gebruik achtereenvolgens "gezond verstand", Depth First Search (DFS) en Breadth First Search (BFS). Wordt het eenvoudiger als alle kosten 1 zijn?
e. Voer het A*-algoritme uit voor onderstaande graaf. Gebruik zonodig de *pathmax equation*. Beginknoop is A, doelknoop is Z. Bij de knopen staat tussen haakjes de (overigens toelaatbare) heuristische functie. Geef duidelijk aan hoe het algoritme verloopt.
f. Wat is de effectieve vertakkingsgraad (= *effective branching factor*) b^* in dit geval? Geef de betreffende formule.

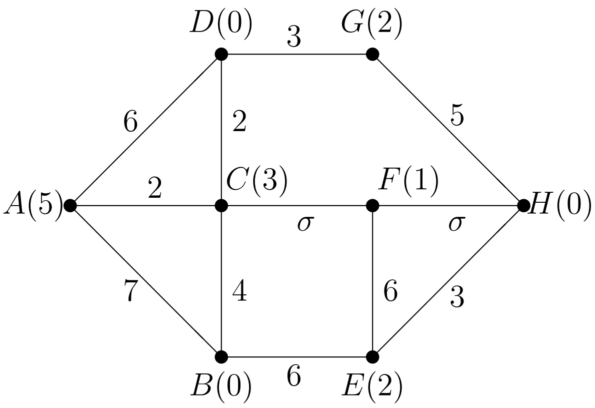


Opgave 9. (opgave van tentamen 25 juni 2003)
a. Leg het A*-algoritme en het IDA*-algoritme uit.

b. Stijgen bij het A*-algoritme de f-waarden langs de paden altijd? Zo nee, wat kun je hieraan doen?
c. Voer het A*-algoritme uit voor onderstaande graaf; hierbij is k een vast positief geheel getal, minstens 2. Gebruik zonodig de pathmax equation. Beginknoop is A, doelknopen zijn Y en Z. Bij de knopen staat tussen haakjes de (overigens admissibele) heuristische functie. De kostenfunctie staat naast de takken van de graaf; tussen de B_i-knopen zit steeds gewicht 1, tussen de C_i's steeds 2 en tussen de D_i's steeds 3. Geef duidelijk aan hoe het algoritme verloopt, en met name in welke volgorde de knopen ontwikkeld worden.
d. Leg uit waarom in dit geval het IDA*-algoritme niet zo prettig werkt.

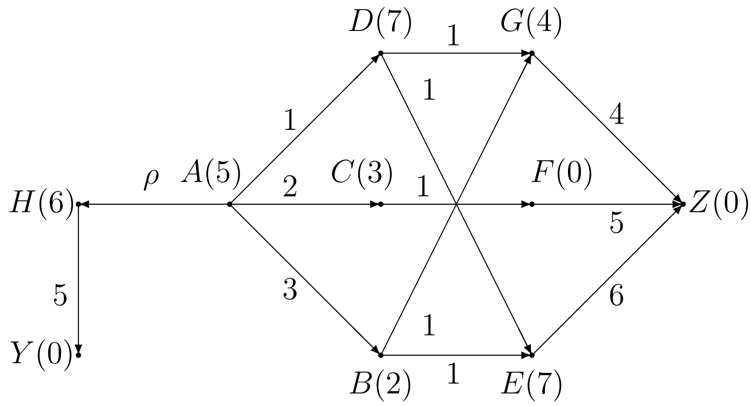


Opgave 10. (opgave van tentamen 4 juni 2004)
a. Leg het A*-algoritme en het IDA*-algoritme uit. Geef expliciet de formule voor f en denk aan de stop-conditie. Geef aan waarin A* en IDA* verschillen.
b. Stijgen bij het A*-algoritme de f-waarden langs de paden altijd? Zo nee, wat kun je hieraan doen?
c. Bekijk onderstaande ongerichte graaf. Beginknoop is A, doelknoop is H. Bij de knopen staat tussen haakjes de heuristische functie. De kostenfunctie staat naast de takken van de graaf. Geef aan voor welke waarden van $\sigma > 0$ de heuristiek admissibel is.

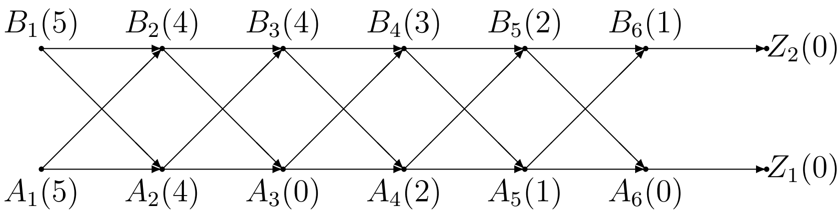


d. Voer het IDA*-algoritme uit voor deze graaf. Neem aan dat σ zo is dat de heuristiek admissibel is. Gebruik zonodig de pathmax equation. Geef duidelijk aan hoe het algoritme verloopt, en met name in welke volgorde knopen ontwikkeld worden. Dit hangt af van de waarde van variabele σ!

Opgave 11. (opgave van tentamen 21 augustus 2006)
a. Leg het A*-algoritme en het IDA*-algoritme uit. Geef expliciet de formule voor f en denk aan de stop-conditie. Geef aan waarin A* en IDA* verschillen.
b. Bekijk onderstaande gerichte graaf. Beginknoop is A, doelknopen zijn Y en Z. De kostenfunctie staat naast de pijlen in de graaf. Bij de knopen staat tussen haakjes de heuristische functie. Maak deze met zo min mogelijk wijziging(en) *admissibel*.
c. Voer het IDA*-algoritme uit voor deze graaf. Gebruik zonodig de *pathmax equation*. Geef duidelijk aan hoe het algoritme verloopt, en met name in welke volgorde knopen ontwikkeld worden. Geef de verschillende mogelijkheden, afhankelijk van $\rho > 0$ (een reëel getal).
d. Geef de best denkbare admissibele heuristische waarden bij de knopen.

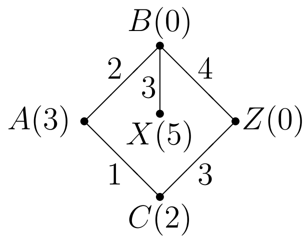


Opgave 12. (opgave van tentamen 22 juni 2011)
a. Leg het A*-*algoritme* en het IDA*-*algoritme* uit. Geef de verschillen duidelijk aan. Geef expliciet de formule voor f (wat stellen g en h voor?) en denk aan de stop-conditie.
b. We bekijken onderstaande *gerichte* graaf. Beginknoop is A_1, doelknopen zijn Z_1 en Z_2. Bij de knopen staat tussen haakjes de waarde van de admissibele heuristische functie h. Alle horizontale takken hebben kosten 1, alle diagonale takken kosten 2. Voer het A*-algoritme uit. Gebruik zonodig de pathmax equation. Geef duidelijk aan hoe het algoritme verloopt, en met name in welke volgorde de knopen ontwikkeld worden. Indien er hierbij keuzes mogelijk zijn, geef ze dan allemaal.



c. Idem, maar nu het IDA*-algoritme.
d. Hoeveel knopen worden er in totaal *ontwikkeld* bij **c** (eventueel afhankelijk van de gekozen volgorde)? En hoeveel als de heuristische h-waarde in alle A-knopen met 1 wordt opgehoogd? Kan het nog beter?

Opgave 13. (opgave van tentamen 13 juni 2012)



a. Leg het A*-*algoritme* en het IDA*-*algoritme* uit. Geef de verschillen duidelijk aan. Geef expliciet de formule voor f (wat stellen g en h voor?) en denk aan de stop-conditie.
b. We bekijken nevenstaande *ongerichte* graaf. Beginknoop is A, doelknoop is Z. Bij de knopen staat tussen haakjes de waarde van de admissibele heuristische functie h.
Voer het A*-algoritme uit. Gebruik zonodig de pathmax equation. Geef duidelijk aan hoe het algoritme verloopt, en met name in welke volgorde de knopen ontwikkeld worden. Indien er hierbij keuzes mogelijk zijn, geef ze dan allemaal.
c. Idem, maar nu het IDA*-algoritme.
d. Nu gaan we weer van A naar Z, maar moeten intussen in X een pakje ophalen. Wat zijn nu toestanden (hint: er zijn er 9 of 10)? Hoe moeten we de heuristische functie h uitbreiden? Geef de best mogelijke h, en laat zien hoe IDA* in het gunstigste geval verloopt.

Opgave 14. (opgave van tentamen 4 juni 2004)

1	2	3
4	5	6
7	8	9

We bekijken het nevenstaande spel voor twee personen. Speler **A** kiest een getal en streept dit getal (bijvoorbeeld 3) en getallen er recht onder en rechts ervan (6 en 9) weg; dit moeten er samen minstens 3 zijn, dus bijvoorbeeld 8 of 9 mogen niet als eerste zet. Daarna doet Speler **B** analoog (bijvoorbeeld 1, en ook 2, 4 en 7 verdwijnen; 5 en 8 blijven). De som van de overgebleven getallen ($5 + 8 = 13$) is de uitkomst van het spel. Speler **A** wil uiteindelijk zo hoog mogelijk eindigen, speler **B** zo laag mogelijk — of juist andersom (%).
a. Geef de *spelboom* (= *game tree*) die hierbij hoort.
b. Beschrijf in woorden het *minimax-algoritme*.
c. Voer dit uit voor de spelboom van **a**, voor beide opties bij (%).
d. Voer het α-β-*algoritme* uit voor beide opties. Geef ook een korte rechtvaardiging voor het snoeien. Zorg ervoor dat de ordening van de knopen zo is dat er in beide gevallen zoveel mogelijk gesnoeid kan worden!

Opgave 15. (opgave van tentamen 19 juni 2007)

We spelen het volgende tweepersoons spel. Speler A kiest een getal, zeg a, uit $\{1, 2, 3, 4\}$. Daarna kiest speler B een getal, zeg b, uit $\{0, 1, 2, 3, 4\}$, maximaal gelijk aan a. Als beide getallen even zijn of oneven, wint A, en anders B. Men wint met $|a - b| + 1$ punten. Aan het begin van het spel mag B kiezen of A zijn keuze zelf mag bepalen of met een eerlijke vierzijdige dobbelsteen.
a. Geef de *spelboom* (= *game tree*) die hierbij hoort. Denk aan kansknopen.
b. Beschrijf in woorden het *minimax-algoritme*.
c. Voer het *expecti-minimax-algoritme* uit voor de spelboom van **a**.
d. Nu mag B aan het begin niets kiezen, A mag geheel zelf zijn keuze bepalen, en B moet $b \neq 0$ nemen. Geef opnieuw de spelboom en voer het minimax-algoritme uit.
e. Voer het α-β-*algoritme* uit, in de situatie van **d**. Zorg ervoor dat de ordening van de knopen zo is dat er zoveel mogelijk gesnoeid kan worden!

Opgave 16. (opgave van tentamen 22 juni 2011)
Cindy en Bert spelen een tweepersoons spel; Cindy begint, daarna moet Bert, en is het spel afgelopen. Er ligt op tafel een rijtje van vier tegels met een letter (de beginletters van de twee spelers) en een getal erop: $\boxed{\text{C3}}$ $\boxed{\text{C5}}$ $\boxed{\text{B4}}$ $\boxed{\text{B8}}$. De speler die aan de beurt is moet naar keuze de voorste *of* de achterste tegel verwijderen. De einduitslag wordt op grond van de twee overblijvende tegels bepaald: als beide tegels dezelfde letter hebben, wint de speler met die beginletter, met de som van de getallen als beloning; verschillen de letters, dan wint het hoogste getal met als beloning het absolute verschil tussen de twee getallen: bij $\boxed{\text{C5}}$ $\boxed{\text{B8}}$ wint Bert met $8 - 5 = 3$ punten. Bert mag —als hij dat wil— direct vóór zijn beurt de *dobbeljoker* inzetten, die de middelste van de drie tegels met de voorste of achterste verwisselt, beide met even grote kans.
a. Beschrijf in woorden het *expecti-minimax-algoritme*.
b. Maak de spelboom, en bereken de expecti-minimax-waarde.
c. Cindy mag van te voren één van de vier getallen met 2 ophogen of verlagen. Wat doet ze als ze verstandig is?
d. Nu wordt er helemaal niet meer gedobbeld, maar mag Bert naar keuze de middelste met de voorste of achterste wisselen; hij mag het ook laten. Voer het *α-β-algoritme* uit. Geef ook een korte rechtvaardiging voor het snoeien. Zorg ervoor dat de ordening van de knopen zo is dat er zoveel mogelijk gesnoeid kan worden!

Opgave 17. (opgave van tentamen 13 juni 2012)
Bekijk het volgende tweepersoons spel, gespeeld door John en Yoko. Ze zijn om en om aan de beurt, en John begint. Er wordt gespeeld met letters uit de verzameling $V = \{A, B\}$, en een 2×2 bord. De speler die aan de beurt is mag kiezen: hij/zij verwijdert een letter van het bord, óf hij/zij zet een letter uit V op een leeg vakje van het bord. Letters uit V mogen maar één keer gebruikt worden. Het spel eindigt zodra de tweede letter uit V gebruikt is. De eerste zet betreft dus het spelen van een letter; deze dient linksboven gezet te worden. De uitslag van het spel wordt als volgt bepaald. Als er alleen één letter rechtsonder staat wint John als het een A is en Yoko als het een B is. Verder wint John als er een lege rij is (en geen lege kolom), Yoko wint als er een lege kolom is (en geen lege rij). Anders is het remise. Men wint steeds met als aantal punten het aantal letters op het bord. Twee voorbeeldspellen:

```
 - -    A -    A -                 - -    B -    - -    - -
 - -    - -    B -   Yoko wint (2) - -    - -    - -    - A   John wint (1)
```

In de *kansversie* van het spel wordt de plek van de gekozen letter random bepaald (voor de beginzet blijft dit steeds linksboven); in de *gewone versie* moeten de spelers zelf de plaats van de gekozen letter bepalen.
a. Beschrijf in woorden het *expecti-minimax-algoritme*.
b. Maak de spelboom en bereken de expecti-minimax-waarde, voor de kansversie.
c. Maak de spelboom en bereken de minimax-waarde, voor de gewone versie.
d. Nu spelen we net als in **c** de gewone versie, zonder kansen. Voer het *α-β-algoritme* uit. Geef ook een korte rechtvaardiging voor het snoeien. Zorg ervoor dat de ordening van de knopen zo is dat er zoveel mogelijk gesnoeid kan worden!