Opgaven Kunstmatige intelligentie — 3
april/mei 2012

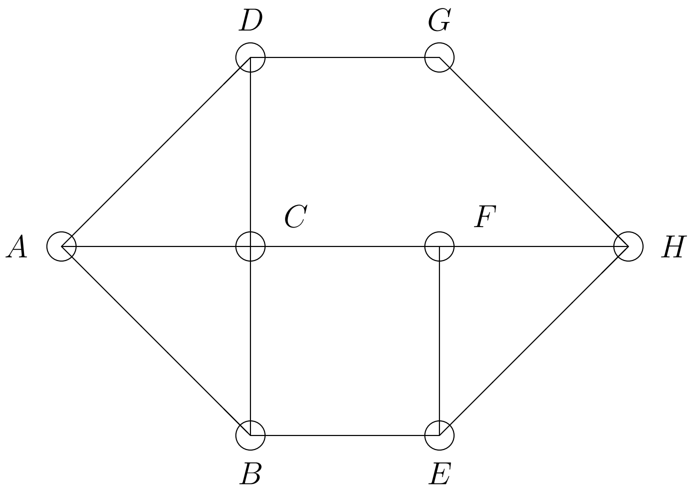
Opgave 20. (opgave van tentamen 25 juni 2008)

We spelen het volgende tweepersoons spel met vier munten, met waarden 10, 20, 30 en X cent, met $X > 30$. Speler A krijgt random een van deze vier munten, met waarde a. Daarna kiest speler B een munt uit de drie overgebleven munten, met waarde b. De hoogste waarde wint. Men wint met $|a - b|$ centen.

a. (6 punten) Geef de *spelboom* (= *game tree*) die hierbij hoort. Denk aan kansknopen.
b. (4 punten) Beschrijf in woorden het *expecti-minimax-algoritme*.
c. (6 punten) Voer het *expecti-minimax-algoritme* uit voor de spelboom van **a**.
d. (4 punten) Nu mag A aan het begin zelf kiezen. Verder geldt $X = 50$. Geef opnieuw de spelboom en voer het minimax-algoritme uit.
e. (5 punten) 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 21. (opgave van tentamen 4 juni 2004)

We willen de knopen in de onderstaande graaf met maximaal 3 kleuren zo kleuren dat aangrenzende knopen verschillend gekleurd zijn. We moeten uiteraard zo weinig mogelijk kleuren gebruiken.



a. Formuleer dit als een *Constraint Satisfaction Probleem.*
b. Leg uit hoe de "most constrained variable" (= "minimum remaining values") heuristiek werkt.
c. Leg uit hoe de "most constraining variable" heuristiek werkt, en geef aan waar deze in het algemeen goed gebruikt kan worden.
d. Leg uit hoe de "least constraining value" heuristiek werkt.
e. Kleur de graaf van de vorige opgave. Maak hierbij van alle drie de heuristieken minstens één maal verstandig gebruik, en geef duidelijk aan wanneer welke methode benut wordt.

Opgave 22.

a. Los de "`TWO + TWO = FOUR`" puzzel op, zie de sheets.
b. Moeten de constraints $\mathtt{F} \neq 0$ en $\mathtt{T} \neq 0$ er bij?

Opgave 23.

Stel dat bij een *Constraint Satisfaction Probleem* de variabelen X_1, X_2 en X_3 hetzelfde domein $D = \{R, G, B\}$ hebben. We willen voor de variabele X_1 de waarde R verbieden, maar mogen geen unaire constraints gebruiken, alleen binaire, en ook de domeinen niet wijzigen. Hoe doen we dat?

Opgave 24.

Probeer een Japanse puzzel (Nonogram) als een *Constraint Satisfaction Probleem* te beschrijven. Een Japanse puzzel is een m bij n rechthoek, waarbij de vakje wit of zwart moeten worden. Naast alle rijen en kolommen staat wat de lengtes van de aaneengesloten series zwarte vakjes zijn. Zo betekent $3, 1, 2$ dat in de betreffende rij of kolom eerst 0 of meer witte vakjes komen, dan precies 3 zwarte, dan 1 of meer witte, 1 zwarte, 1 of meer witte, 2 zwarte en tot slot 0 of meer witte — in die volgorde.
Helpt het als je hier hulpvariabelen introduceert?

Opgave 25. (opgave van tentamen 13 augustus 2001)

We willen met behulp van een *neuraal netwerk* op grond van tijdstip, temperatuur en precieze locatie een voorspelling maken voor windrichting en waterhoogte in een haven. Daartoe maken we een netwerk met drie invoerknopen, drie verborgen knopen, en vijf uitvoerknopen (waaronder vier voor de windrichting: N/O/Z/W).
a. Leg uit waar “bias-kopen” (extra -1-inputs) voor nodig zijn. Ons netwerk krijgt er twee.
b. Teken de netwerkarchitectuur. Geef kort in woorden aan hoe het *BackPropagation* algoritme werkt.
c. Wat heeft *Ockham’s razor* te maken met het aantal verborgen knopen?
d. Er is voor gekozen om de windrichting met vier uitvoerknopen te coderen. Wat is hiervan een nadeel, en geef een andere mogelijkheid.

Opgave 26. (opgave van tentamen 4 juni 2004)

a. Geef een Neuraal Netwerk met twee invoeren en één uitvoer, dat de XOR-functie (de “exclusieve of”) berekent.
b. Waarom kan een netwerk zonder verborgen knopen de functie van **a** niet realiseren? Leg uit.
c. Leid de *Backpropagation* update/leerregel voor een gewicht W_{ij} op de tak van verborgen knoop j naar uitvoerknoop i af. Gebruik leersnelheid α, doelwaarde T_i, net-uitvoer O_i en activatie a_j.
d. Hoe zou je, door deze leerregel eenvoudig aan te passen, er voor kunnen zorgen dat de gewichten zo dicht mogelijk bij 0 komen?

Opgave 27.

a. Geef —analoog aan de Wumpus-wereld— een beschrijving van de *stofzuigerwereld*, met behulp van eenvoudige predicaten-logica. De wereld heeft twee aangrenzende kamers, waar al of niet stof ligt. Stof is alleen in de kamer zelf waarneembaar.
Gebruik bijvoorbeeld de “opdrachten” `Beweeg` en `Zuig`, “toestanden” als `Stof` en `Schoon`, en functies als `BestAction` en `Stoffig`.
b. Wat zijn *effect-* en *frame-axioma’s*? Geef van beide een eenvoudig voorbeeld uit de stofzuigerwereld.