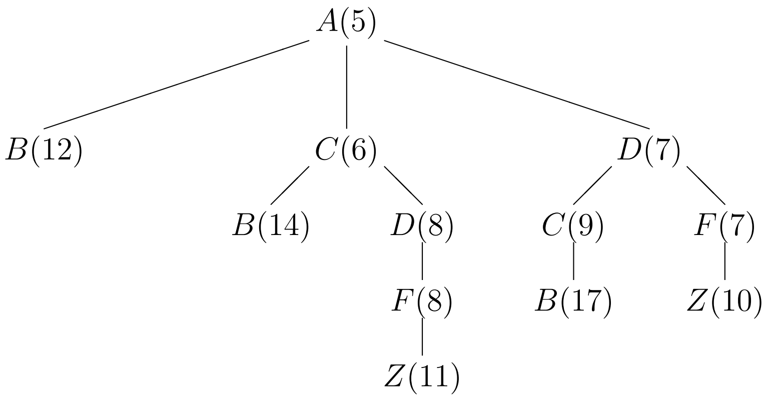
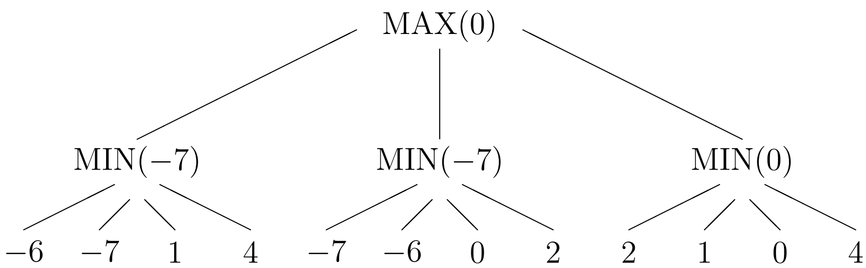
Oude tentamens Kunstmatige intelligentie
Uitwerkingen (selectie)
Universiteit Leiden — Informatica — 2005

Opgave 1. A* (20/100 punten; tentamen 1 juni 2001)
Uiteindelijk wordt onderstaande boom opgebouwd. Knopen die al eerder op een pad voorkwamen worden niet aangegeven, er zijn dus geen kringen (bijvoorbeeld onder $C(6)$ komt niet opnieuw A te hangen). Tussen haakjes staat naast de knopen hun f-waarde. Achtereenvolgens worden geopend = ontwikkeld = geexpandeerd $A(5)$, $C(6)$, $D(7)$, $F(7)$, $D(8)$ (let op, hoewel al een doelknoop gevonden is, worden eerst nog knopen met lagere f-waarde ontwikkeld), $F(8)$, $C(9)$ en tot slot $Z(10)$. Bij $F(8)$ staat dankzij de *pathmax equation* 8 in plaats van 7, bij $F(7)$ staat evenzo 7 in plaats van 6.



De effectieve vertakkingsgraag b^* laat zich berekenen uit $1 + b^* + (b^*)^2 + \cdots + (b^*)^d = N$, met $d = 3$ de diepte van de oplossing, en $N = 8$ het aantal geopende knopen.

Opgave 2. α-β-algoritme (20/100 punten; tentamen 1 juni 2001)
Denk er aan de waardes in de bladeren (in dit geval eindsituaties; in het algemeen wordt hier een evaluatie-functie aangeroepen) goed te "kiezen": als speler 1 de MAX-speler is, geef dan de situaties met uitslag groter dan 5 een min-teken. Het α-β-algoritme snoeit de drie meest rechtse kinderen van de middelste knoop, en dat is te begrijpen: de MAX-speler kan via zijn linker kindknoop al een antwoord -7 bewerkstelligen, dus als (dankzij het meest linkse kind van de middelste knoop) de middelste knoop -7 of nog lager oplevert, heeft het geen zin nog te kijken naar die drie meest rechtse kinderen. De spelboom ziet er als volgt uit:



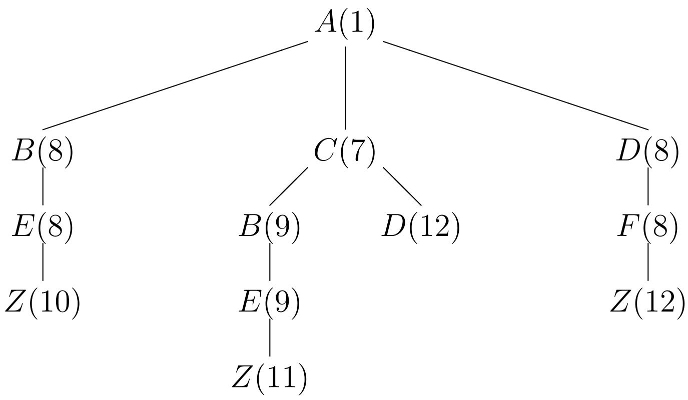
Opgave 3. ID3 (20/100 punten; tentamen 1 juni 2001)
Denk er aan ook te vertellen wat er gebeurt als er geen voorbeelden meer zijn voor een knoop, wat als er geen attributen meer over zijn (kies de meerderheidswaarde, zie ook **b**), en wat als alle voorbeelden dezelfde classificatie hebben (dan heb je een ja- of nee-knoop). Geef ook de formule voor Gain(attribuut).
Voor het probleem met zeer veel mogelijke attribuutwaarden kan bijvoorbeeld de Gain()-waarde gedeeld worden door iets wat de groottes van de verschillende klassen "meet", bijvoorbeeld:

$$\mathit{SplitInfo}(A) = -\sum_{i=1}^{\nu} \frac{|S_i|}{|S|} \log(\frac{|S_i|}{|S|}),$$

als de betreffende voorbeelden S door het attribuut A in ν klassen $S_1, \ldots, S_\nu$ verdeeld wordt.

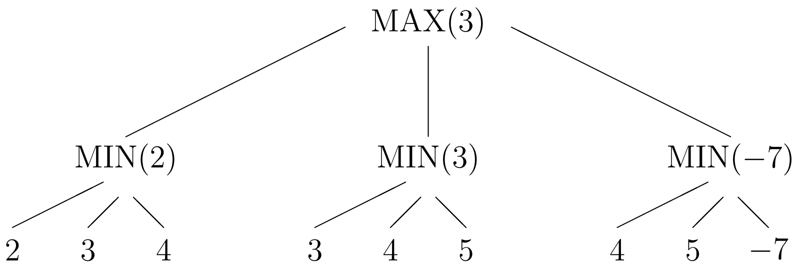
Opgave 4. Genetische algoritmen (20/100 punten; tentamen 1 juni 2001)
Een bitstring moet een compleet rooster coderen. Start bijvoorbeeld met een matrix met als rijen de uren van de week en als kolommen de klassen. Zet per matrixelement een leraar (bijvoorbeeld VRIJ)—even aannemend dat leraren en vakken uniek gekoppeld zijn. Zet deze matrix met rijen achter elkaar in een string. Ook is het mogelijk een rij van tupels (vak,klas,uur,leraar,lokaal) te maken, als rijen maar altijd even lang zijn.
Een evaluatiefunctie moet als het ware een rapportcijfer geven aan zo'n rooster. Er zijn harde constraints: de klassen moeten de hun toebedeelde vakken krijgen, leraren kunnen niet op meerdere plaatsen tegelijk aanwezig zijn, en per lokaal en tijd mag er maximaal één klas zijn. Dit soort conflicten leveren strafpunten op. Evenzo, maar dan minder hard, zijn er (minder) strafpunten voor tussenuren en slechte vakverdelingen. Al deze punten opgeteld geven een evaluatiefunctie.
Met behulp van selectie, crossover en mutatie werkt het Genetisch algoritme zich een weg naar een goede oplossing.
Speciale mutaties kunnen bijvoorbeeld twee leraren verwisselen, of een VRIJ uur toevoegen. Bij crossover kun je met daggrenzen of klassen rekening houden.
Een Genetisch algoritme heet *elitair* als de beste x individuen steeds naar de volgende generatie gaan.

Opgave 5. A* en IDA* (20/100 punten; tentamen 13 augustus 2001)



Uiteindelijk wordt bovenstaande boom opgebouwd. Tussen haakjes staat naast de knopen hun f-waarde. Achtereenvolgens worden geopend = ontwikkeld = geexpandeerd $A(1)$; f-limiet 7: $C(7)$; f-limiet 8: $B(8)$, $E(8)$ (dankzij pathmax van 7 naar 8), $D(8)$, $F(8)$ (dankzij pathmax van 7 naar 8); f-limiet 9: $B(9)$, $E(9)$ (dankzij pathmax van 8 naar 9); f-limiet 10: $Z(10)$, klaar. Let op, hoewel al een doelknoop gevonden is, worden eerst nog knopen met lagere f-waarde ontwikkeld; pas als een doelknoop ontwikkeld wordt, stopt het algoritme.

Opgave 6. α-β-algoritme (20/100 punten; tentamen 13 augustus 2001)
Orden de kinderen van links naar rechts als volgt: voor het linkse pak 1 lucifers, voor het middelste 2 lucifers en voor het rechtse 3 lucifers. De spelboom ziet er als volgt uit:



Het α-β-algoritme snoeit niks. Als je de volgorde van de drie kinderen van de MIN(-7) knoop andersom doet, dus in de volgorde -7, 5, 4, worden de takken naar 4 en 5 gesnoeid. En dat is te begrijpen: de MAX-speler kan via de middelste kindknoop al een antwoord 3 bewerkstelligen, dus als (dankzij het linkse kind van de rechter knoop) de rechter knoop -7 of nog lager oplevert, heeft het geen zin nog te kijken naar die twee meest rechtse kinderen.

Opgave 7. Bayesiaanse netwerken (20/100 punten; tentamen 13 augustus 2001)
Er geldt: $P(Z|O) = P(O|Z)P(Z)/P(O)$ (Bayes). Nu is

$$\begin{aligned} P(O|Z) &= P(O|L,M)P(L|Z)P(M|Z) + P(O|\overline{L},M)P(\overline{L}|Z)P(M|Z) + \\ &\quad P(O|L,\overline{M})P(L|Z)P(\overline{M}|Z) + P(O|\overline{L},\overline{M})P(\overline{L}|Z)P(\overline{M}|Z) \\ &= 0.9 * 0.6 * 0.8 + 0.5 * 0.4 * 0.8 + 0.5 * 0.6 * 0.2 + 0.0 * 0.4 * 0.2 = 0.652. \end{aligned}$$

Analoog $P(O|\overline{Z}) = \ldots = 0.197$, en dus $P(O) = P(O|Z)P(Z) + P(O|\overline{Z})P(\overline{Z}) = 0.334$, waaruit volgt: $P(Z|O) = 0.652 * 0.3/0.334 \approx 0.59$.
Er geldt: **b** is diagnostisch en **c** causaal.

Opgave 8. Neurale netwerken (20/100 punten; tentamen 13 augustus 2001)
Als de uitvoer 4 knopen voor de windrichting heeft kan het gebeuren dat ze alle vier bijvoorbeeld 1 geven: wat betekent dat? Handiger wellicht één knoop die de windrichting met één getal codeert, bijvoorbeeld de hoek.
Verder:

$$\frac{\partial}{\partial W_{j,i}} \quad 1/2 \sum_t Err_t^2 = -Err_i \, a_j \, g'(in_i),$$

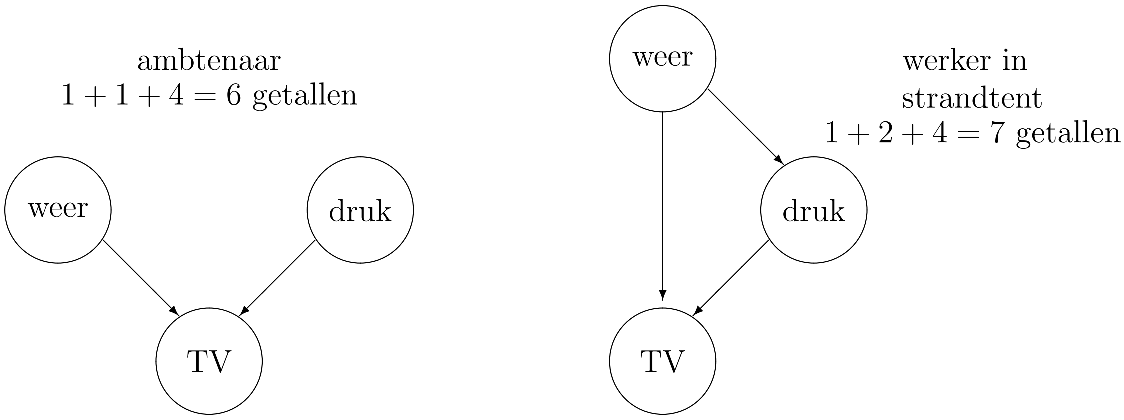
want $Err_i = T_i - O_i = T_i - g(in_i)$ en $in_i = \sum_\ell W_{\ell,i} a_\ell$; tot slot: α is de leersnelheid. En de gradiënt wijst in de richting van de grootste fout-stijging, dus er komt nog een extra – bij als je een dalende fout wilt.

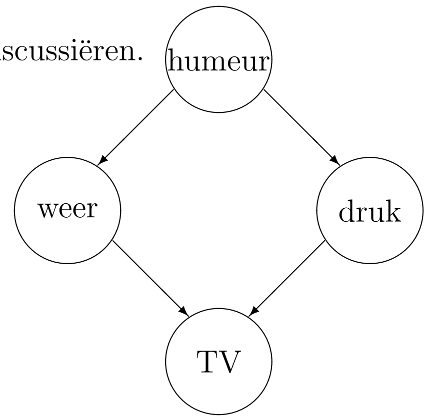
Opgave 9. A*/IDA* (20/100 punten; tentamen 5 juni 2002)

Opgave 10. α-β-algoritme (20/100 punten; tentamen 5 juni 2002)
De wortelknoop krijgt uiteindelijk de waarde 5. Hij heeft een kind (een blad met tak gelabeld "stop"), met waarde 5, en een kind gelabeld "dobbel". De knoop daaronder heeft vier kinderen, met kansen 1/4 op de takken, met waardes 3, 4, 6 en 8, en krijgt

zelf de waarde $(3+4+6+8)/4 = 5.2$. Bijvoorbeeld de knoop met 4 erin heeft weer een stop-knoop en een dobbelknoop, waaronder 1, 5 en 6 hangen.

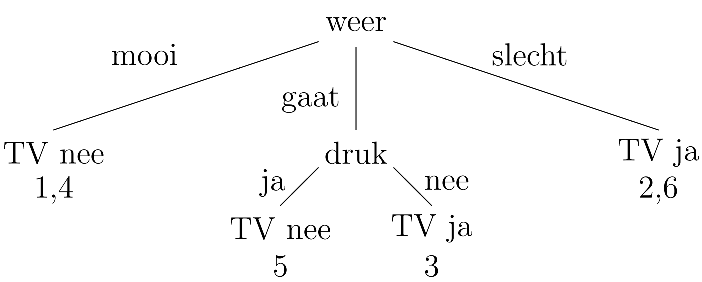
Opgave 11. Bayesiaanse netwerken (20/100 punten; tentamen 5 juni 2002)

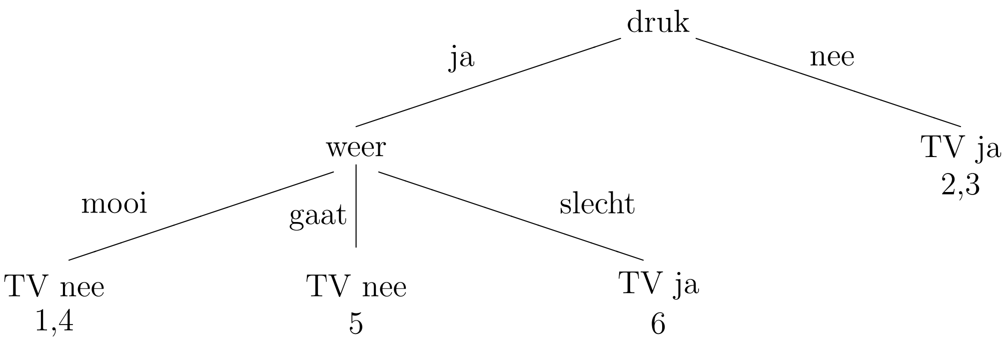




Over de tak van weer naar TV (rechtsboven) valt te duiscussiëren.
Diagnostisch P(mooi weer | TV ja),
causaal P(TV nee | mooi weer),
intercausaal P(mooi weer | TV ja $\wedge$ wel druk)
(alle drie voor ambtenaar),
mixed P(niet druk | TV ja $\wedge$ slecht weer)
(voor werker in strandtent).
Met "cutset" humeur: maak twee netwerken, één voor humeur goed en één voor humeur slecht — zonder kringen.

Opgave 12. ID3 (20/100 punten; tentamen 5 juni 2002)





$I(3/6, 3/6) = 1$, de entropie vooraf. Na de eerste weer-vraag:

$$1/3 \times I(1,0) + 1/3 \times I(1/2,1/2) + 1/3 \times I(0,1) = 1/3 \times 0 + 1/3 \times 1 + 1/3 \times 0 = 1/3,$$

met gain: $1 - 1/3 = 2/3$.

Opgave 13. Neurale netwerken (20/100 punten; tentamen 5 juni 2002)
$W_{j,i} \leftarrow W_{j,i} + \alpha \times I_j \times \Delta_i$ met $\Delta_i = (T_i - O_i) \cdot g'(in_i)$, want $\frac{\partial E}{\partial W_{j,i}} = -(T_i - O_i) \cdot g'(in_i) I_j$. Let er op dat er een extra min bijkomt omdat je tegen de gradiënt in wilt: je wilt juist een zo laag mogelijk fout.

Opgave 14. A*/IDA* (20/100 punten; tentamen 12 augustus 2002)

Opgave 15. α-β-algoritme (20/100 punten; tentamen 12 augustus 2002)

Opgave 16. ID3 (20/100 punten; tentamen 12 augustus 2002)

Opgave 17. Genetische algoritmen (20/100 punten; tentamen 12 augustus 2002)

Opgave 18. A*/IDA* (20/100 punten; tentamen 25 juni 2003)
De route langs de D's kost $3k$, die langs de C's $2k + 3$ en die langs de B's kost $k + 8$. Dus voor $k < 3$: D-route, $k = 3$: C of D, $k = 4$: C, $k = 5$: C of B, en $k > 5$: B-route.
d. Veel verschillende f-waarden, zodat je vaak opnieuw een depth-first wandeling moet maken.

Opgave 19. α-β-algoritme (20/100 punten; tentamen 25 juni 2003)
De wortelknoop krijgt minimax-waarde 3.25.

Opgave 20. Bayesiaanse netwerken (20/100 punten; tentamen 25 juni 2003)
a. $P(w)$, $P(a|w)$, $P(a|\overline{w})$, $P(e|w,a)$, $P(e|\overline{w},a)$, $P(e|w,\overline{a})$, $P(e|\overline{w},\overline{a})$, $P(b|a)$, $P(b|\overline{a})$, $P(k|a,e)$, $P(k|\overline{a},e)$, $P(k|a,\overline{e})$ en $P(k|\overline{a},\overline{e})$
b. $P(B|A) = P(B|A,W)$
c. $P(b|\overline{w}) = P(b|a)P(a|\overline{w}) + P(b|\overline{a})P(\overline{a}|\overline{w})$ (en $P(\overline{a}|\overline{w}) = 1 - P(a|\overline{w})$).
d. $P(k|\overline{w}) = P(k|a,e)P(e|\overline{w},a)P(a|\overline{w}) + P(k|\overline{a},e)P(e|\overline{w},\overline{a})P(\overline{a}|\overline{w})$
$\quad + P(k|a,\overline{e})P(\overline{e}|\overline{w},a)P(a|\overline{w}) + P(k|\overline{a},\overline{e})P(\overline{e}|\overline{w},\overline{a})P(\overline{a}|\overline{w})$
e. $P(a|\overline{b},w) = P(\overline{b}|a,w)P(a|w)/P(\overline{b}|w) = P(\overline{b}|a)P(a|w)/P(\overline{b}|w)$ (via Bayes, b en c)
f. c en d causaal, e mixed

Opgave 21. Theorie (diversen) (20/100 punten; tentamen 25 juni 2003)

Opgave 22. A*/IDA* (20/100 punten; tentamen 4 juni 2004)
b. Neem $\sigma \geq 3/2$ (dan admissibel).
c. Bij het ontwikkelen van C–5 kom je F–$3 + \sigma$ tegen. Als $\sigma \leq 2$ wordt dit dankzij "pathmax" natuurlijk 5. Let er tevens op dat als $2 < \sigma < 3$ bij f-limiet 6 ook knoop F ontwikkeld wordt (anders alleen B).

Opgave 23. α-β-algoritme (20/100 punten; tentamen 4 juni 2004)
a. Je krijgt bladeren 9 (enig kind), 15, 10 (2 broers), 12, 13, 11 (3 broers), 10, 17 (2 broers), 9 (enig kind), 11, 11 en 6 (3 broers).
c,d. Voor **A** hoog: 11, met maximaal 4 "geprunede" bladeren; voor **A** laag: 9, met maximaal 6 "geprunede" bladeren. (Dat komt omdat de 11 steeds 2 broers heeft, en 9 enig kind is.)

Opgave 24. Neurale Netwerken (20/100 punten; tentamen 4 juni 2004)

d. Ockham prefereert "korte" (goede) verklaringen boven "lange" (goed) verklaringen. Dus liever minder verborgen knopen.

Opgave 25. Constraint Satisfaction Problemen (20/100 punten; tentamen 4 juni 2004)
e. Gebruik de "most constraining variable" heuristiek voor de keus van de eerste te kleuren knoop: kies C, want die heeft de meeste buren: 4 stuks. Kies dan bijvoorbeeld A. Nu hebben B en D nog maar één mogelijke kleur, en dus kleuren we die nu (volgens de "minimum remaining values" heuristiek. Kies vervolgens knoop E. Volgens de "least constraining value" heuristiek geven we die dezelfde kleur als C. De rest is eenvoudig te kleuren. (Er zijn andere oplossingen mogelijk.)

Opgave 26. Theorie (diversen) (20/100 punten; tentamen 4 juni 2004)