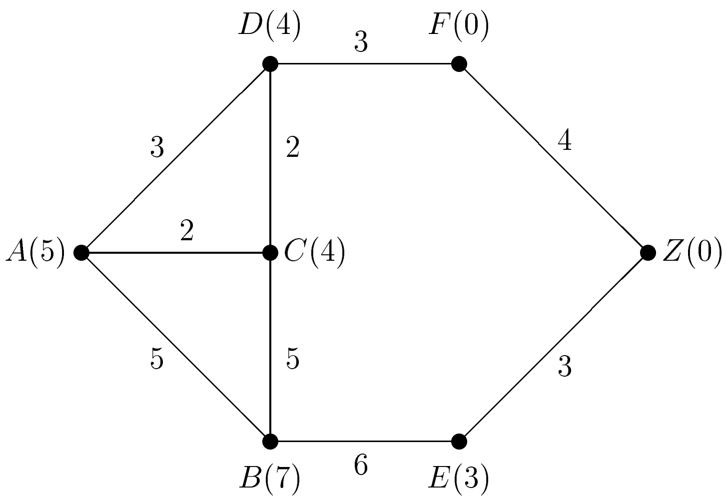
Tentamen Kunstmatige intelligentie
Universiteit Leiden — Informatica
vrijdag 1 juni 2001, 14.00–17.00 uur

Geef bij alle antwoorden een korte en duidelijke toelichting. Veel succes!

Opgave 1. A* (20 punten)

a. (5 punten) Leg het A*-algoritme uit.
b. (2 punten) Wanneer heet een heuristiek *toelaatbaar* (= admissible)?
c. (4 punten) Wat is de *pathmax equation* en waarvoor wordt deze gebruikt?
d. (5 punten) 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. De kostenfunctie staat naast de takken van de graaf. Geef duidelijk aan hoe het algoritme verloopt.



e. (4 punten) Wat is de effectieve vertakkingsgraad (= *effective branching factor*) b^* in dit geval? Geef de betreffende formule.

Opgave 2. α-β-algoritme (20 punten)

We bekijken het volgende spel voor twee personen. Het bestaat uit een drie bij drie bord met daarop de getallen 1 tot en met 9, en wel als volgt gerangschikt:

3	8	9
2	4	6
5	1	7

De eerste speler kiest een getal. De tweede speler kiest nu een getal dat in een andere rij en een andere kolom moet staan. De uitslag van het spel wordt bepaald door het getal uit de enige rij en kolom die nog nooit zijn aangewezen. Als dat lager dan 5 is heeft de eerste speler gewonnen, is het hoger dan 5 dan heeft de tweede speler gewonnen, en als het 5 is is het remise. Een voorbeeld: de eerste speler kiest 8, de tweede 5; de uitslag van het spel wordt bepaald door het getal 6: de tweede speler wint, en wel met 6 (het overblijvende getal) flippo's.

a. (5 punten) Geef het gedeelte van de spelboom (= game tree) dat hoort bij de situatie waarbij de eerste speler een getal uit de eerste rij (dus 3, 8 of 9) kiest. Broerknopen worden in oplopend gesorteerde volgorde genoteerd.
b. (5 punten) Beschrijf in woorden het *minimax-algoritme*.
c. (5 punten) Voer dit uit voor de spelboom van **a**.
d. (5 punten) Voer ook het α-β-*algoritme* uit voor deze spelboom, en geef duidelijk aan waar eventueel gesnoeid (= "gepruned") wordt. Geef ook een korte rechtvaardiging voor dit snoeien.

Opgave 3. ID3 (20 punten)
We hebben een database waarin een aantal voorbeelden aan de hand van een stel attributen beschreven worden; verder hebben we een Ja/Nee-classificatie voor deze voorbeelden.
a. (10 punten) Leg uit hoe het *ID3-algoritme* werkt. Bij de uitleg mag vrijelijk gebruik gemaakt worden van de entropie $I(p/(n+p), n/(n+p))$ (p is het aantal positieve voorbeelden, n het aantal negatieve), zonder deze nader uit te leggen of te definiëren.
b. (5 punten) Geef een mogelijke oplossing voor het probleem dat zich voordoet als er bij het splitsen elkaar tegensprekende voorbeelden zijn.
c. (5 punten) Geef een mogelijke aanpassing van het algoritme die voorkomt dat bij het splitsen attributen met zeer veel mogelijke waardes (zoals de datum) gekozen worden.

Opgave 4. Genetische algoritmen (20 punten)
Stel dat we een lesrooster voor een school moeten maken, en dat we dat met een *genetisch algoritme* willen doen.
a. (5 punten)
Geef (in woorden) globaal aan hoe een eenvoudige codering van de oplossingen er uit zou kunnen zien.
b. (5 punten) Geef (in woorden) aan hoe een evaluatiefunctie (= fitnessfunctie) er uit zou kunnen zien.
c. (5 punten) Geef de drie standaard operatoren van een genetisch algoritme, waarbij één van de drie ook speciaal op dit probleem toegesneden moet zijn. Leg ze kort uit.
d. (5 punten) Wanneer heet een genetisch algoritme *elitair*?

Opgave 5. Data mining (20 punten)
We bekijken de volgende transactie-database met verkoopgegevens over de producten (= items) `1,2,...,10`:

```
1: 3,4,7,9
2: 3,6,8,9,10
3: 3,5,7,9
4: 1,3,8
5: 2,3,9,10
```

Klant `1` bijvoorbeeld heeft producten `3`, `4`, `7` en `9` gekocht.
a. (5 punten) Geef de definitie van *support* en *frequente itemset*.
b. (5 punten) Geef de frequente itemsets voor de database uit het voorbeeld; gebruik een minimale support (= *support threshold*) van 2.
c. (10 punten) Leg uit hoe het *Apriori-algoritme* deze itemsets weet te vinden. Laat zien welke itemsets hierbij gesnoeid (= "gepruned") worden.