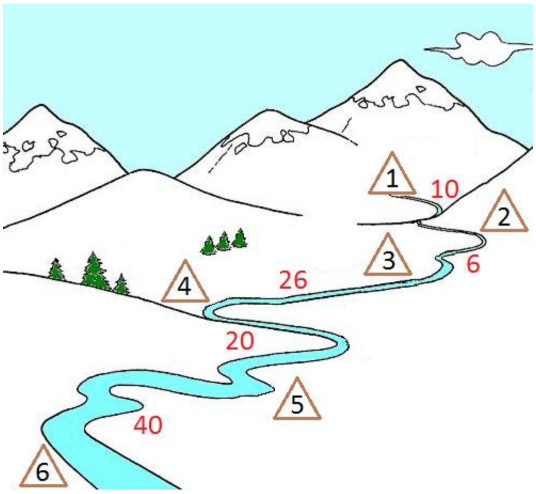
5

Voorbeeldexamenvragen

5.1 **Schuilplaatsen bouwen**

(a) De kajakclub heeft zes kampeerplaatsen langs de rivier in de bergen, zie onderstaande figuur. De figuur toont de reistijden (in minuten) van de ene kampeerplaats naar de andere, wanneer de kajakkers de rivier afdalen van de bergen naar de vallei. Om de rivier naar boven te bevaren, neemt de tijd echter met de helft toe. Dus, kajakken van plaats 1 naar plaats 2 duurt 10 minuten, kajakken van plaats 2 naar plaats 1 duurt 15 minuten.



De kajakclub wil de kampeerplaatsen uitbouwen tot echte schuilplaatsen, waar alle kajakkers kunnen komen schuilen als het stormt en onweert. Maar ze hebben slechts voldoende bouwmaterialen om 3 van de 6 kampeerplaatsen om te bouwen tot schuilplaats. Bij slecht weer zullen de kajakkers die op de andere plaatsen kamperen naar de dichtstbijzijnde schuilplaats moeten varen, m.a.w. deze die ze in de kortste tijd kunnen bereiken.
De kajakclub heeft de drie schuilplaatsen zo gekozen dat bij storm de maximale tijd T die nodig is om een schuilplaats te bereiken van de andere drie kampeerplaatsen *zo kort mogelijk* is. Wat is deze maximale tijdsspanne T? Verklaar.

15 minuten	16 minuten	20 minuten	24 minuten	26 minuten	30 minuten

(b) Beschouw het volgende algemene probleem:
Gegeven een padgraaf $P = (a_1, \ldots, a_n)$, met een kost $c_{i,i+1}$ voor elke voorwaartse

boog (a_i, a_{i+1}) en een kost $c'_{i+1,i}$ voor elke teruggaande boog (a_{i+1}, a_i). Gegeven een waarde k voor het aantal te selecteren toppen.
Beschouw een selectie S van k toppen. Voor elke niet-geselecteerde top j bepalen we de gewogen afstand naar de dichtstbijzijnde geselecteerde top, noem dit d_j. Het maximum van deze waarden noteren we door $T_S(n, k)$.
Gevraagd is de selectie S^* te bepalen die $T_{S^*}(n, k)$ minimaliseert. Geef een efficiënt algoritme voor dit probleem. Leg uit waarom je algoritme correct is.

5.2 **Vertegenwoordigers kiezen**
Gegeven is een groep G van n personen $(x_1, \ldots, x_n)$ en een reeks van m kennisrelaties binnen deze groep, van de vorm $\{x_i, x_j\}$, hetgeen aangeeft dat x_i en x_j elkaar kennen.
Het is de bedoeling binnen deze groep een deelgroep D van vertegenwoordigers te kiezen, zodanig dat iedereen een vertegenwoordiger kent of er zelf een is. M.a.w. van elke kennisrelatie behoort minstens één van de personen tot de deelgroep van vertegenwoordigers. Deze groep van vertegenwoordigers moet zo klein mogelijk zijn.

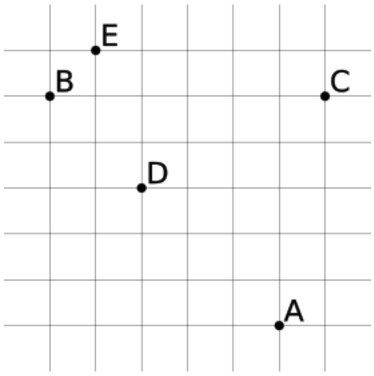
(a) Beschouw het volgende gretige algoritme: Kies een persoon x die zoveel mogelijk andere personen kent en voeg die toe aan D. Verwijder alle kennisrelaties die x bevatten. Herhaal dit tot er geen kennisrelaties meer te beschouwen zijn.
Geef een tegenvoorbeeld waarvoor dit algoritme niet gegarandeerd de kleinste groep vertegenwoordigers vindt.

(b) Beschouw het volgende gretige algoritme: Kies een kennisrelatie $e = (x, y)$ die nog geen vertegenwoordiger bevat. Voeg x en y toe aan D. Verwijder alle relaties die x bevatten en alle relaties die y bevatten. Herhaal tot er geen relaties meer zijn.
Bewijs dat de gevonden vertegenwoordigersgroep hoogstens tweemaal zo groot is als de minimum vertegenwoordigersgroep, m.a.w. dat dit algoritme een benaderingsverhouding 2 heeft.

5.3 **Softwarelicenties aankopen**
Je wilt een beveiligingsbedrijf opstarten, en hiervoor moet je n licenties van cryptografische software aankopen. Hierbij geldt echter de regel dat je slechts één dergelijke licentie per maand mag aankopen.
Elke licentie kost op dit moment 100 euro, maar hun prijzen stijgen exponentieel. In het bijzonder stijgt de prijs van licentie j elke maand met een factor $r_j > 1$, met r_j een gegeven parameter. Dit betekent dat, als je licentie j pas over t maanden aankoopt, je hiervoor $100 \times r_j^t$ euro zult betalen. We veronderstellen dat alle prijsstijgingsfactoren verschillend zijn, m.a.w. dat $r_i \neq r_j$ voor licenties $i \neq j$.
Gevraagd is te bepalen in welke volgorde je de licenties moet aankopen, zodanig dat de totale aankoopprijs zo klein mogelijk is. Geef een algoritme met polynomiale uitvoeringstijd voor dit probleem. Argumenteer waarom je algoritme correct is.

5.4 **Een vergaderplaats zoeken**

(a) Vijf bevriende studenten Alain, Beatrijs, Christophe, Dorelinda en Engelbert wonen op de plaatsen aangeduid met de letters A, B, C, D en E in onderstaande figuur.



Ze willen een vergaderplaats op een kruispunt afspreken om samen te studeren.
Ze wandelen naar de vergaderplaats via de straten (zwarte lijnen) en elke student zal de afstand afmeten aan het totale aantal blokken horizontaal en verticaal vanaf hun eigen huis. Bijvoorbeeld, de afstand van Alains huis naar Engelberts huis is 10 blokken.
Wat is de kortst mogelijke totale afstand van de vijf huizen tot aan de vergaderplaats? Verklaar.

17 blokken	18 blokken	19 blokken	20 blokken

(b) Beschouw het volgende algemene probleem:
Gegeven een $m \times m$ rooster, waarbij de lengte van alle blokken gelijk is aan 1. De afstand tussen twee kruispunten in het rooster wordt gedefinieerd als de Manhattan-afstand, m.a.w. het totale aantal blokken horizontaal en verticaal tussen de twee punten.
Gegeven n verschillende kruispunten $(p_1, \ldots, p_n)$ in het rooster. Gevraagd is het kruispunt p te bepalen waarvoor de som van de afstanden van elk van de punten p_j tot p zo klein mogelijk is.
Geef een efficiënt algoritme voor dit probleem. Leg uit waarom je algoritme correct is.

5.5 **Maximum onafhankelijke verzameling in een boom**
Zij gegeven een graaf $G = (V, E)$. Een onafhankelijke verzameling S is een deelverzameling $S \subset V$ van paarsgewijs niet-adjacente toppen. Een maximum onafhankelijke deelverzameling is een onafhankelijke deelverzameling met maximum cardinaliteit.
In algemene grafen is het bepalen van een maximum onafhankelijke verzameling een onhandelbaar probleem, maar voor bomen kan het probleem wel in polynomiale tijd opgelost worden.
Geef een algoritme met polynomiale uitvoeringstijd voor het bepalen van een maximum onafhankelijke verzameling in een boom. Argumenteer waarom je algoritme correct is.