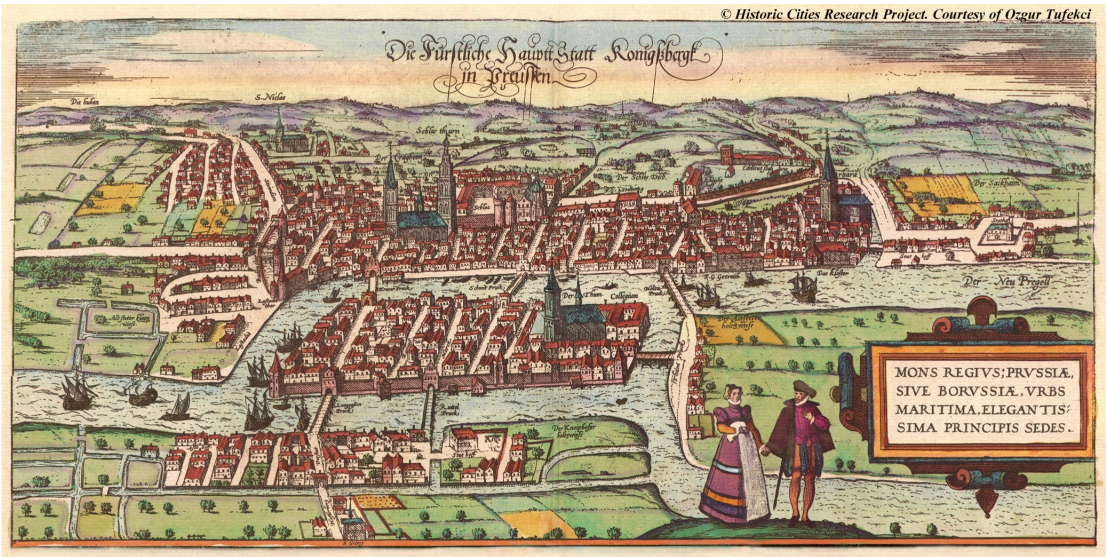
Module 1

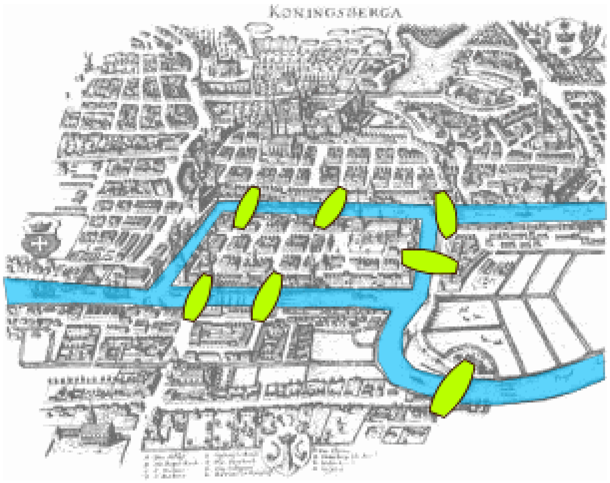
Ontstaan van grafen



1.1 De zeven bruggen van Königsberg

Het volgende historische verhaal illustreert dat ook een ludiek en een op het eerste zicht banaal praktisch probleem, voorgelegd aan een wiskundig brein, tot een fundamenteel nieuwe theorie leidt met een breed spectrum aan nuttige toepassingen.

De stad Königsberg, het huidige Kaliningrad aan de Oostzee, ligt aan de samenvloeiing van twee rivieren, de oude en de nieuwe Pregel, die de stad verdeelden in vier van elkaar gescheiden regio's, waaronder het eiland Kneiphof. In de 18de eeuw waren de verschillende oevers onderling verbonden door zeven bruggen, zoals aangegeven in onderstaande bewerkte kaart van Königsberg in de 18de eeuw.



Volgens de legende brachten de inwoners van het toen florerende Königsberg hun zondagmiddag al wandelend door in hun mooie stad. Tijdens het kuieren beslisten de inwoners om een uitdaging voor henzelf in het leven te roepen. Hun doel was om **een wandeling te maken doorheen de stad zodat elke brug precies eenmaal wordt overgestoken, ongeacht of men al dan niet op de oever van vertrek weerkeert**. Hierbij wordt stilzwijgend verondersteld dat een stadsdeel enkel vanuit een ander stadsdeel kan worden bereikt via een brug, en eens een brug opgelopen, mag men niet halverwege rechtsomkeer maken.

Niemand van de inwoners kon een wandeling maken die aan deze voorwaarde voldoet. Van waar men ook startte en welke keuzes men bij de verschillende brugaansluitingen ook maakte, alle pogingen bleken zonder succes. Maar ze konden ook niet aantonen dat een dergelijke route niet bestaat. Gelukkig voor hen, was Königsberg niet ver weg van Sint-Petersburg, de plaats waar de later wereldberoemde wiskundige Leonard Euler werkte.

Waarom zou de toen 28-jarige Euler zich bezighouden met het oplossen van dit probleem dat ondertussen tot een publieke passie was uitgegroeid? Euler was druk bezig met het publiceren van boeken en artikels (meer dan 500 gedurende zijn leven). Hij schreef niet alleen over wiskunde, maar ook over mechanica, optica, astronomie, navigatie en hydrodynamica.

Daarom was het niet verrassend dat Euler het bruggenprobleem in Königsberg als triviaal bestempelde, zoals bleek uit een brief in 1735 aan Carl Leonhard Gottlieb Ehler, burgemeester van Danzig, die hem had gevraagd om het probleem op te lossen (vrij vertaald naar het Nederlands):

"... Dus je ziet, meest edele Heer, hoe dit soort oplossing weinig verband heeft met wiskunde, en ik begrijp niet waarom je verwacht van een wiskundige dit te doen, meer dan van eender wie, aangezien de oplossing enkel op rede steunt, en de ontdekking niet afhangt van een wiskundige methode. Daarom, zie ik niet in waarom problemen die zo weinig verband met wiskunde hebben sneller door wiskundigen zouden worden opgelost dan door anderen."

Toch was Euler geïntrigeerd door het bruggenprobleem zoals bleek uit een brief in datzelfde jaar aan een andere wiskundige en ingenieur Giovanni Marinoni (vrij vertaald naar het Nederlands):

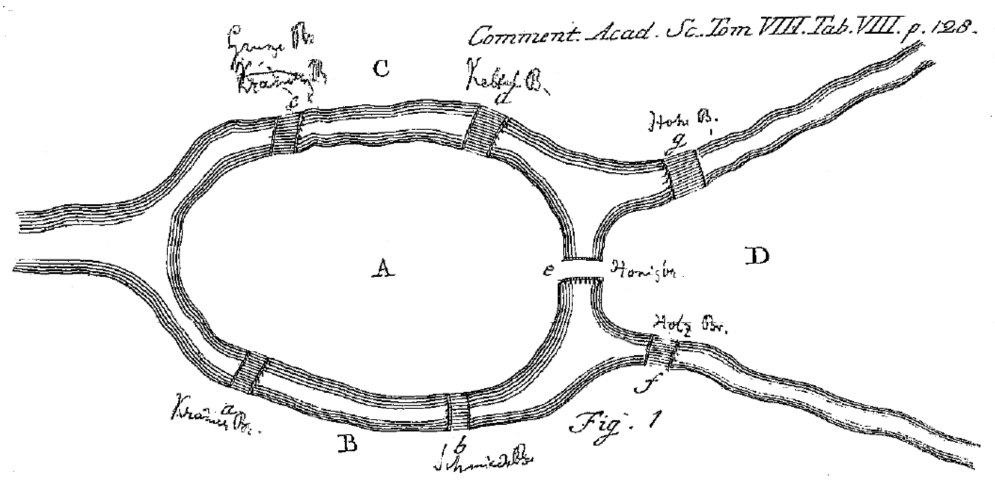
"Deze vraag is zo banaal, maar lijkt mijn aandacht waardig omdat noch meetkunde, noch algebra, zelfs niet de kunst van tellen, volstonden om dit op te lossen."

Euler zag in dat dit probleem verwant was aan een onderwerp dat Gottfried Wilhelm Leibniz ooit had besproken en benoemd als *geometria situs*, een nieuwe 'meetkunde van situaties'.

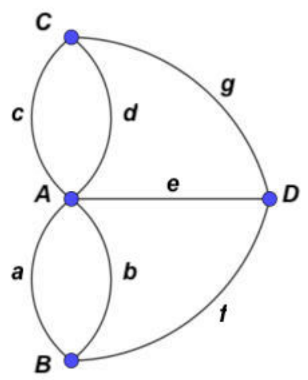
In 1735 presenteerde Euler een artikel, genoemd 'Solutio problematis ad geometriam situs pertinentis' waarin niet enkel het probleem van de zeven bruggen van Königsberg is opgelost, maar ook de veralgemening van dit probleem voor elk aantal gescheiden stadsdelen en elk aantal bruggen. Na een grondige analyse en een abstracte benadering van het probleem kon hij bewijzen dat een dergelijke wandeling in Königsberg onmogelijk was. De wijze waarop hij dit deed is het historisch startpunt van de nieuwe 'meetkunde van situaties' dat nu uitgegroeid is tot de grafentheorie.

1.2 Abstractie van het bruggenprobleem van Königsberg

Het is evident dat het panorama van de stadsdelen en de architectuur van de bruggen geen enkele rol spelen voor de oplossing van het probleem. De eerste stap die Euler zette, was het probleem mathematiseren en voorstellen als een meer abstracte veralgemeenbare vorm. Zie Euler's Figuur 1 uit Solutio problematis ad geometriam situs pertinentis:



De stadsdelen krijgen een hoofdletter als naam (A, B, C, D) en de bruggen een kleine letter (a, b, c, d, e, f).
In een verdere vorm van abstrahering worden de stadsdelen als punten en de bruggen als kromme of rechte lijnstukken voorgesteld.



Een dergelijke voorstelling wordt een graaf genoemd. De punten worden knopen (of toppen) genoemd. De kromme of rechte lijnstukken die knopen verbinden, worden bogen genoemd. Deze graaf is een voorbeeld van een multigraaf omdat sommige knopen door meer dan één boog met elkaar verbonden zijn.

Terminologie en definities

Een **graaf** bestaat uit punten die wel of niet met elkaar verbonden zijn. De punten noemen we **knopen** (of toppen). De verbindingen tussen knopen worden **bogen** genoemd. In een **enkelvoudige graaf** worden elke twee knopen met elkaar verbonden door hoogstens één boog. Zijn er knopen die door meer dan één boog met elkaar verbonden worden, dan spreekt men van een **multigraaf**.

Formeel wiskundig wordt een graaf als volgt gedefinieerd.

Wiskundige definitie

Een (enkelvoudige) **graaf** G is een koppel (V, E), waarbij V een niet-lege verzameling is bestaande uit **knopen** en E een verzameling bestaande uit paren (of doubletons) van knopen uit V. *Een paar (of doubleton) is een verzameling met 2 elementen.*
De elementen van E noemen we **bogen**.

Opmerking: V en E komen uit het Engels: 'vertices' zijn hoekpunten en 'edges' zijn randen of kanten.

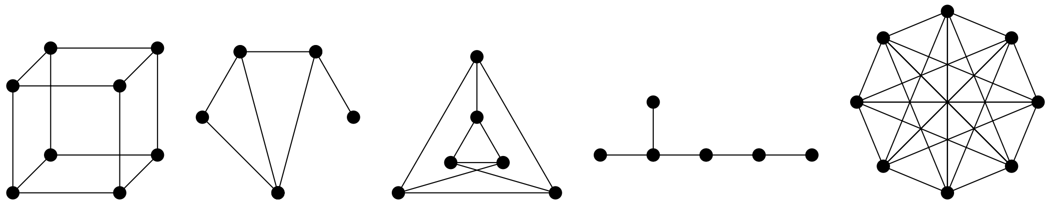
Voor het vervolg hebben we nog meer definities nodig.

Definities

We noemen het aantal knopen in een graaf G of dus het aantal elementen van de verzameling V de **orde** van de graaf G.
Het aantal bogen of dus het aantal elementen van E is de **grootte** van de graaf G.

Bovenstaande definities smeken om visualisatie.
In een tekening worden de knopen van een graaf voorgesteld door punten (of cirkeltjes) met de regel dat twee punten (of cirkeltjes) worden verbonden door een (kromme) lijn als die twee knopen een boog vormen.

Voorbeelden orde en grootte van een graaf:

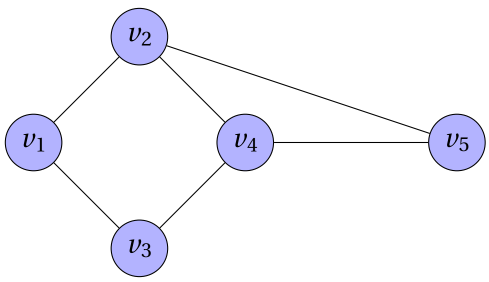


orde **8** grootte **12** | orde **5** grootte **6** | orde **6** grootte **9** | orde **6** grootte **5** | orde **8** grootte **20**

Notatie

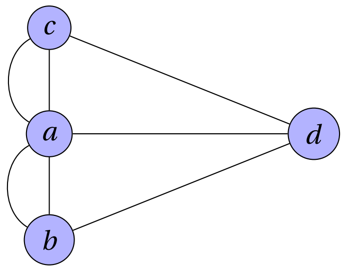
Zowel knopen als bogen worden genoteerd d.m.v. kleine letters, eventueel m.b.v. indices. Afhankelijk van de context, worden soms hoofdletters gebruikt om knopen te benoemen.
Bogen kunnen ook worden genoteerd a.d.h.v. de knopen die ze verbinden, bv. de boog die knopen a en b verbindt is $\{a, b\}$. We zullen dit kort noteren als ab.

Voorbeeld 1



De verzameling van de knopen $V = \{v_1, v_2, v_3, v_4, v_5\}$ bevat 5 elementen, de orde van deze graaf is dus 5.
De verzameling van de bogen $E = \{v_1 v_2, v_1 v_3, v_2 v_4, v_2 v_5, v_3 v_4, v_4 v_5\}$ bevat 6 elementen, de grootte van deze graaf is dus 6.

Voorbeeld 2
De graaf horend bij het bruggenprobleem stellen we vanaf nu als volgt voor.

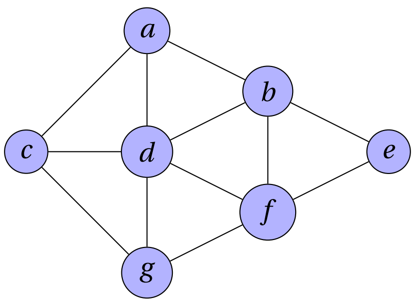


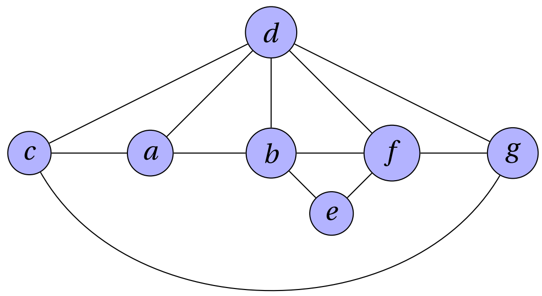
Deze multigraaf heeft orde 4 en grootte 7.

Opmerkingen

1. Als ab een boog in een graaf is, dan is ba dezelfde boog in de graaf, immers $\{a, b\} = \{b, a\}$.

2. Eenzelfde graaf kan op verschillende manieren worden voorgesteld. De plaats van de knopen in het vlak is niet van belang, enkel het aantal knopen en de bogen die de knopen verbinden. Bogen mogen elkaar ook snijden. We proberen wel steeds op een zo overzichtelijk mogelijke manier een graaf voor te stellen. Onderstaande twee figuren stellen dezelfde graaf voor.





Opdracht: teken zelf nog een andere voorstelling van deze graaf.

1.3 Herformulering van het bruggenprobleem van Königsberg

Om het bruggenprobleem van Königsberg te formuleren in de context van grafen, voeren we eerst nog enkele begrippen in.

Definities

Een **wandeling** door een graaf is een rij van aaneensluitende bogen, d.w.z. twee opeenvolgende bogen hebben steeds een uiteinde gemeen.

Een **spoor** is een wandeling waarbij geen bogen meer dan eenmaal worden doorlopen.

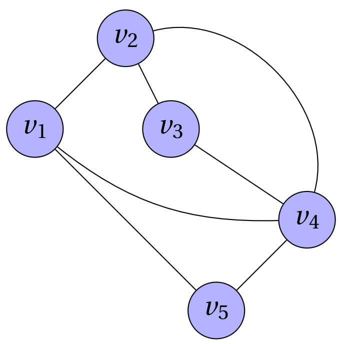
Een graaf is **samenhangend** als er voor elk paar knopen een wandeling van de ene knoop naar de andere bestaat, m.a.w. als elke knoop bereikbaar is vanuit elke andere knoop.

Notatie

Een wandeling W wordt genoteerd als een aantal geordende bogen die opeenvolgend doorlopen worden.
Bij een enkelvoudige graaf kan een wandeling ook genoteerd worden a.d.h.v. de knopen die opeenvolgend verbonden worden.

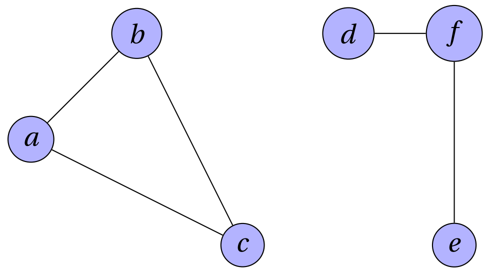
Voorbeelden

1. Beschouw de enkelvoudige graaf G met
   knopen $V = \{v_1, v_2, v_3, v_4, v_5\}$ en
   bogen $E = \{v_1v_2, v_2v_3, v_2v_4, v_3v_4, v_1v_4, v_1v_5, v_4v_5\}$.
   Dus de orde van deze graaf G is 5 en de grootte is 7.

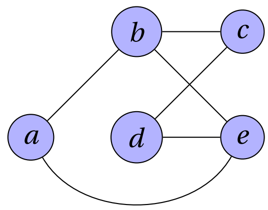
   

   Een wandeling in deze graaf G is bv. $W_1 = (v_1, v_4, v_2, v_3, v_4, v_1, v_5)$.
   Dit is geen spoor omdat boog v_1v_4 tweemaal wordt doorlopen.
   De wandeling $W_2 = (v_1, v_5, v_4, v_2, v_3, v_4)$ is wel een spoor omdat geen enkele boog meer dan eenmaal wordt doorlopen.

2. Onderstaande graaf K is niet samenhangend.
   Er bestaat bv. geen wandeling van a naar f.



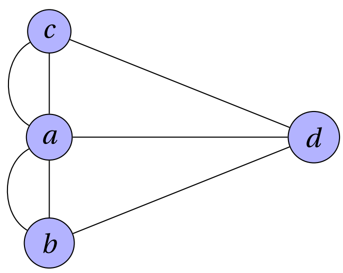
3. In de volgende graaf H is $W = (b, e, d, c, b)$ een spoor die eindigt in het beginpunt. Een dergelijk spoor is gesloten en wordt een **circuit** genoemd.



Hiermee kan het bruggenprobleem van Königsberg worden geformuleerd m.b.v. grafen.

Probleemstelling

Bestaat er een spoor in onderstaande graaf dat alle bogen van deze graaf precies eenmaal doorloopt?



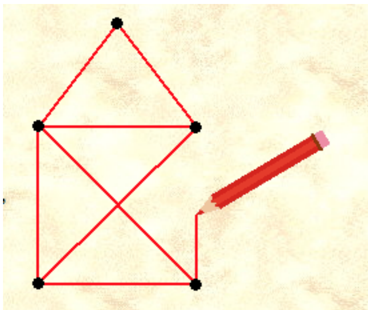
Een dergelijk spoor wordt een euleriaans spoor genoemd.

Definitie

Een **euleriaans spoor** is een spoor die alle bogen van de graaf precies eenmaal doorloopt.

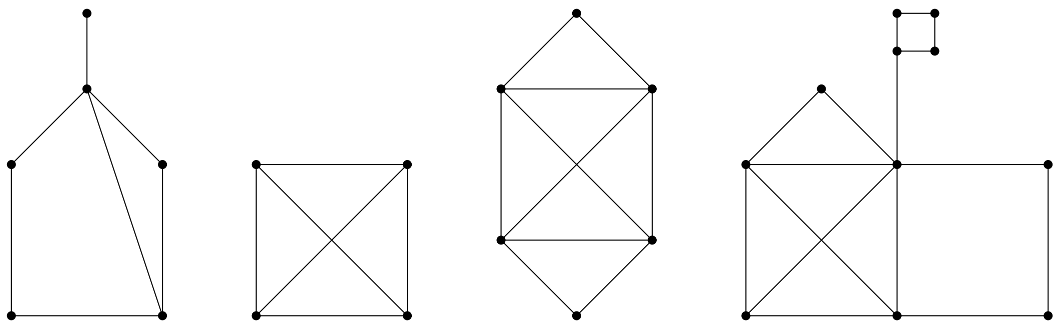
1.4 Euleriaans spoor

Kinderen zoeken zonder het te beseffen euleriaanse sporen. Het spelletje waarbij een huisje moet worden getekend zonder het potlood op te heffen en zonder meermaals over dezelfde lijn te gaan is een poging om een euleriaans spoor te tekenen.



Onderzoeksopdracht 1

1. Ga na dat je inderdaad dit 'kruishuisje' kan tekenen door elke lijn precies eenmaal te doorlopen en zonder je potlood op te heffen.
2. Onderzoek welke van onderstaande figuren je kan tekenen zonder je potlood op te heffen en zonder tweemaal over dezelfde lijn te gaan.



3. Probeer de oorzaak te achterhalen waarom het bij de ene figuur lukt en bij de andere niet.

We voeren nog een nieuw begrip in zodat we de gevonden hypothese meer wiskundig a.d.h.v. grafen kunnen formuleren.

Definities

Het aantal bogen dat in een knoop v vertrekt (of aankomt) noemen we de **graad** van de knoop.

Naburige (of adjacente) knopen zijn twee knopen die verbonden zijn door een boog.

Gevolg

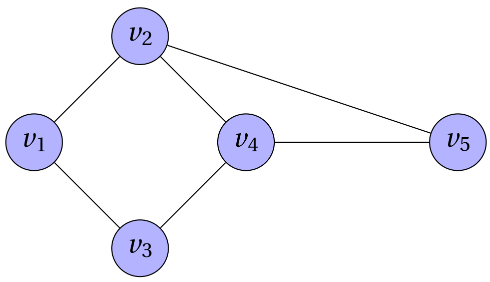
De graad van een knoop in een enkelvoudige graaf is het aantal naburige knopen.

Notatie

De graad van een knoop v wordt genoteerd als $\deg(v)$ (van het Engelse *degree*).

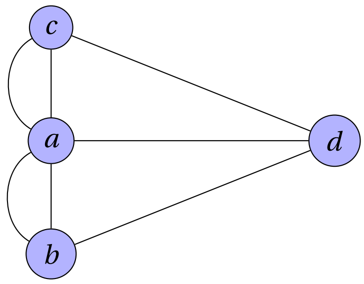
Opmerking: de graad van een knoop is steeds even of oneven.

Voorbeelden



De graad van v_1 is 2, of kort genoteerd: $\deg(v_1) = 2$.
Verder geldt hier: $\deg(v_2) = 3$, $\deg(v_3) = 2$, $\deg(v_4) = 3$, $\deg(v_5) = 2$.
De knopen v_1 en v_2 zijn buren of naburige knopen, v_1 en v_4 zijn geen buren.

We bekijken opnieuw de graaf horend bij het bruggenprobleem.



In deze multigraaf is bv. $\deg(a) = 5$.
Enkel de knopen b en c zijn niet naburig.

Opdracht
Formuleer nu zelf opnieuw de hypothese opdat een graaf een euleriaans spoor bevat.

Stelling

Een samenhangende graaf bezit een euleriaans spoor als en slechts als hij nul of twee knopen met oneven graad heeft.

Euler bewees de implicatie ($\Rightarrow$), met andere woorden het feit dat de voorwaarde in verband met de graden van de knopen nodig is.

Bewijs $\Rightarrow$
Veronderstel een samenhangende graaf met een euleriaans spoor.
In elke knoop waarin je passeert, kom je toe via een boog en vertrek je via een andere boog. De graad van een dergelijke knoop is steeds even. Voor begin- en eindknoop zijn er twee mogelijkheden:

- ofwel zijn begin- en eindknoop verschillend en dus beide van oneven graad,
- ofwel vallen begin- en eindknoop samen en zijn dus beide van even graad.

De omgekeerde stelling ($\Leftarrow$) is pas in de 19de eeuw bewezen door Carl Hierholzer.

Onderzoeksopdracht 2

1. Ga nu zelf na of een wandeling in Königsberg mogelijk is waarbij elke brug precies eenmaal wordt doorlopen.
2. Kan je door een brug bij te bouwen wel een euleriaans spoor bekomen?

1.5 Euleriaans circuit

We kunnen aan dergelijke problemen een extra eis toevoegen: begin- en eindknoop moeten samenvallen. In dit geval zijn we op zoek naar een gesloten euleriaans spoor of euleriaans circuit.

Definities

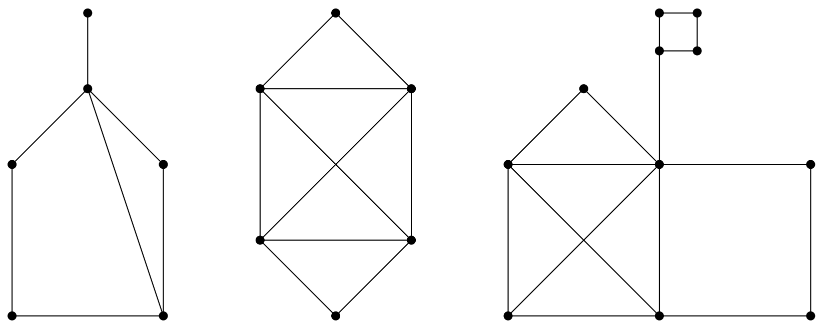
Een gesloten spoor of **circuit** is een spoor dat vertrekt en eindigt in dezelfde knoop.

Een gesloten euleriaans spoor of **euleriaans circuit** is een circuit waarbij elke boog in de graaf precies éénmaal wordt doorlopen.

Een **eulergraaf** is een graaf die een euleriaans circuit bevat.

Opdrachten

1. We bekijken de grafen uit de eerste onderzoeksopdracht opnieuw. Welke van onderstaande grafen is een eulergraaf?



2. Pas de formulering van de stelling omtrent de nodige en voldoende voorwaarde voor een euleriaans spoor aan zodat je een euleriaans circuit bekomt.

Stelling

Een samenhangende graaf is een eulergraaf als en slechts als de graad van elke knoop even is.

Bewijs

$\Rightarrow$ Als de graaf euleriaans is, dan moet er een euleriaans circuit bestaan. Voor elke keer dat het euleriaans circuit langs een boog toekomt in een knoop, moet het opnieuw vertrekken via een andere boog grenzend aan deze knoop. Of anders gezegd: voor elke 'inkomende' boog, moet er een 'uitgaande' boog zijn. Na het eindigen in de startknoop heeft het circuit een even aantal bogen per knoop aangedaan. Gezien alle bogen behoren tot het euleriaans circuit, zijn de graden van alle knopen even.

$\Leftarrow$ TB: als in een samenhangende graaf alle knopen een even graad hebben, dan bestaat er een euleriaans circuit.
We bewijzen dit door een algoritme op te stellen waarmee we zo'n ciruit bouwen.

- Stap 1: We kiezen een willekeurige startknoop v.
- Stap 2: We starten de wandeling vanuit die knoop via een boog. Dit is altijd mogelijk, de graaf is immers samenhangend. De startknoop kan, evenals elke andere knoop, niet geïsoleerd zijn.
- Stap 3: We komen in een volgende knoop toe.
- Stap 4: We zetten de wandeling verder via een andere boog. Dit is mogelijk, aangezien de graad van elke knoop even is. Dus voor elke 'toekomende' boog is er ook een 'vertrekkende' boog.
- We herhalen stap 3 en stap 4 tot we terug in de startknoop zijn beland.

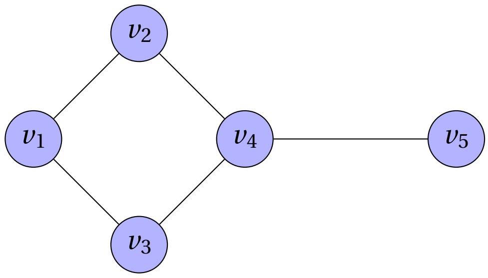
Nu zijn er twee mogelijkheden:

1. Het doorlopen circuit bevat alle bogen van de graaf en is dus euleriaans. Daarmee is de kous af.
2. Het doorlopen circuit bevat nog niet alle bogen van de graaf. In dit geval breiden we het gevonden circuit als volgt uit. We doorlopen het circuit vanaf de startknoop en houden halt aan de eerste knoop w waar een nog niet gebruikte boog vertrekt. We breken het bestaand circuit open en volgen de nieuwe boog, waarna we bovenstaande stappen 3 en 4 toepassen, maar dan enkel met niet gebruikte bogen. Dit is opnieuw mogelijk wegens de even graden (voor elke toekomende boog is er een vertrekkende boog). We herhalen deze stappen tot we terug in w zijn aanbeland, daarna vervolgen we onze wandeling via het eerder geconstrueerd circuit. Het nieuwe bekomen circuit is ofwel euleriaans ofwel nog niet. In het laatste geval kan de vorige constructie herhaald worden. Wegens het samenhangend zijn (elke knoop en boog is bereikbaar) en aangezien het aantal bogen eindig is, moet deze herhaalde constructie hoe dan ook tot een euleriaans circuit leiden. $\square$

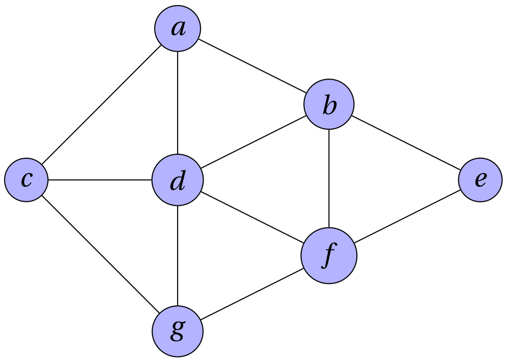
Onderzoeksopdracht 3
Kan je door één of meerdere bruggen bij te bouwen een euleriaans circuit bekomen in Königsberg?

1.6 Oefeningen

1. Gegeven:
   Graaf 1



   Graaf 2



   Bepaal voor beide grafen:

	orde	grootte	graad van elke knoop
Graaf 1			
Graaf 2			

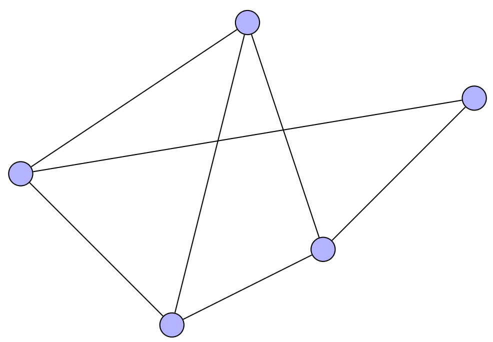
2. Teken grafen 3,4, en 5 die voldoen aan volgende eigenschappen.
   Zijn deze grafen uniek of zijn er meerdere grafen die voldoen aan de gegevens?
   Vul de tabel verder aan.

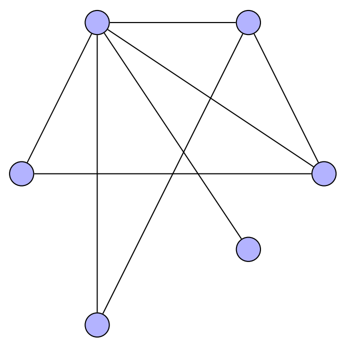
	orde	grootte	graad van elke knoop
Graaf 3	6	8	4,4,3,2,2,1
Graaf 4	5		3,3,2,2,2
Graaf 5			5,5,3,3,2,2,0

3. Het aantal bogen van graaf 5 is 10. Je zal geen graaf kunnen tekenen met meer of minder bogen die beantwoordt aan de opgave. Er is immers een verband tussen het aantal bogen en de graden van de knopen van de graaf.

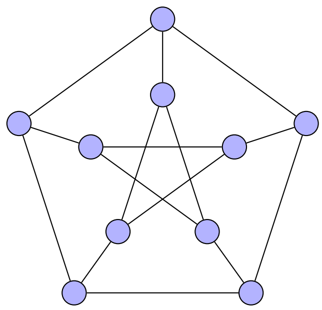
   (a) Wat is het verband tussen de som van de graden van alle knopen en het aantal bogen?
   (b) Toon dit verband aan.
   (c) Hieruit volgt een eigenschap over de som van de graden van alle knopen. Formuleer deze eigenschap.
   (d) Wat kan je zeggen over het aantal knopen met oneven graad in een graaf? Verklaar je antwoord.

4. Snijdende bogen in een graaf geven vaak een onduidelijk beeld van een graaf. Teken voor onderstaande grafen een voorstelling zonder snijdende bogen.



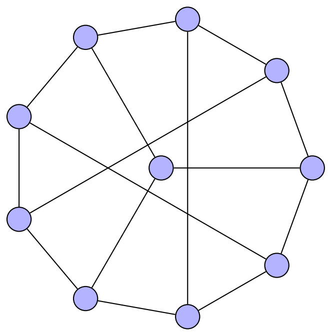


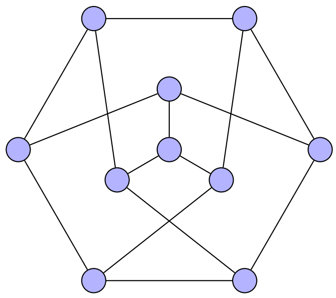
5. Een bekende en belangrijke graaf die op de meest onverwachte plaatsen opduikt in de grafentheorie is de petersengraaf, vernoemd naar de Deen Julius Petersen.
   Een voorstelling van de graaf is



   Bepaal van deze graaf de orde, de grootte en de graad van de knopen.

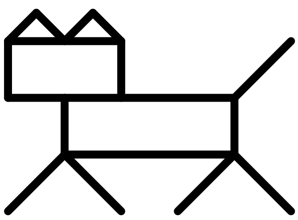
6. Hieronder zie je nog twee voorstellingen van grafen.



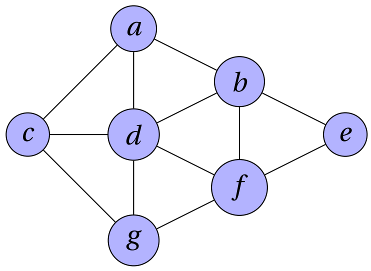


   Zijn dit ook voorstellingen van de petersengraaf. Zo ja, waarom? Zo neen, waarom niet?

7. Hoeveel keer moet je minstens je potlood opheffen om onderstaande figuur te tekenen? Verklaar je antwoord.
   Als je dit dier twee ogen en een mond wil geven, hoeveel keer moet je dan je potlood opheffen?

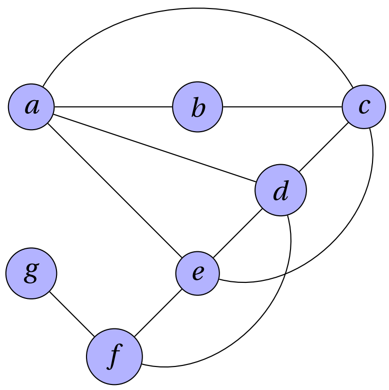


8. Beschouw opnieuw graaf 2 uit oefening 1.



   (a) Geef een wandeling in deze graaf die geen spoor is.
   (b) Pas de wandeling aan zodat je een spoor bekomt.
   (c) Kan je een euleriaans spoor vinden? Verklaar je antwoord.
   (d) Pas de graaf aan door een minimum aantal bogen bij te tekenen zodat je een enkelvoudige eulergraaf bekomt.
   (e) Pas de graaf aan door een minimum aantal bogen bij te tekenen zodat je een eulergraaf bekomt die een multigraaf is.

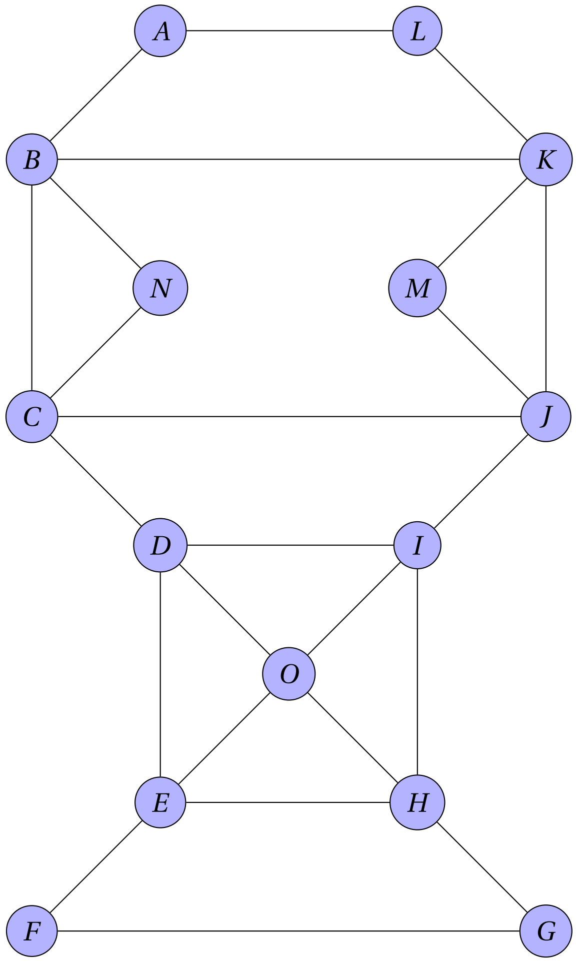
9. Beschouw onderstaande graaf.



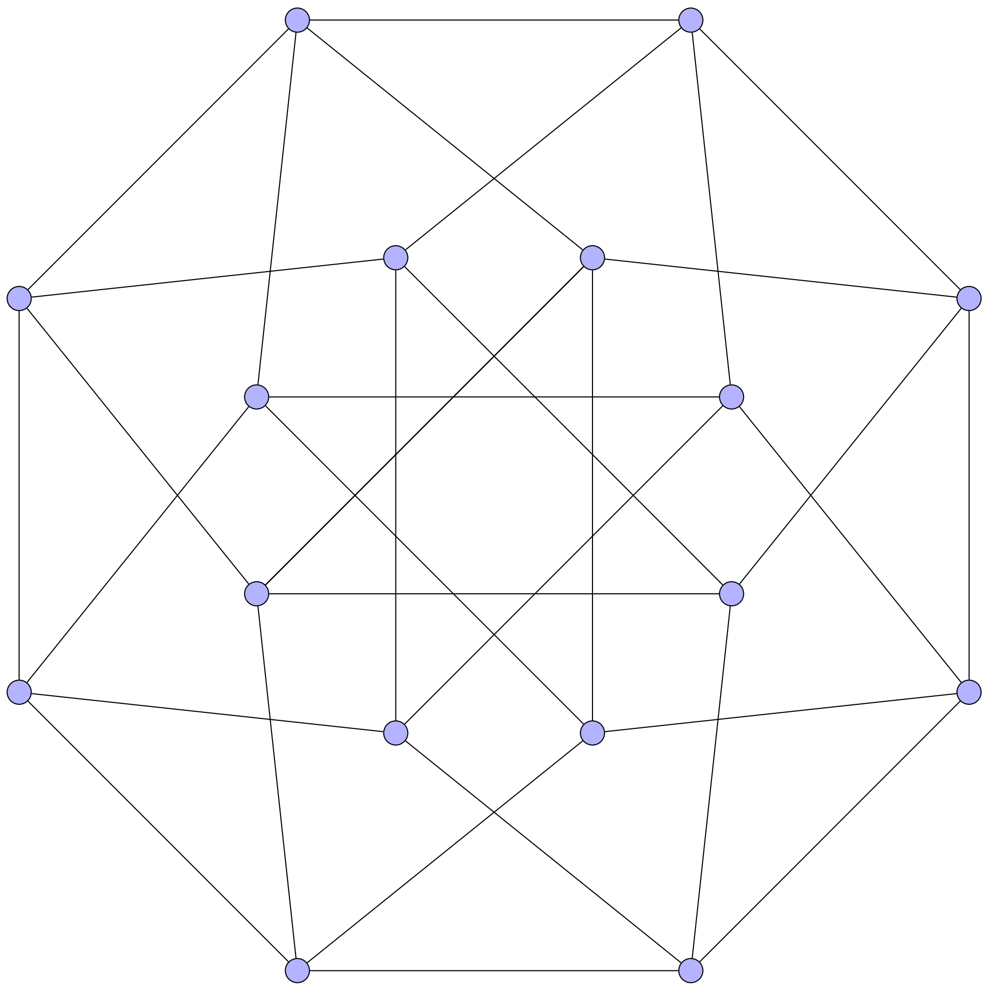
   Bepaal voor deze graaf

   (a) een voorstelling zonder snijdende bogen.
   (b) een euleriaans spoor. Als dat niet mogelijk is, hoeveel bogen moet je dan minimaal weglaten om een euleriaans spoor te krijgen? Pas de graaf aan en geef dit spoor.
   (c) een euleriaans circuit. Als dat niet mogelijk is, hoeveel bogen moet je dan minimaal toevoegen om een euleriaans circuit te krijgen? Pas de graaf aan zodat het een enkelvoudige graaf blijft en geef dit circuit.

10. Je werkt bij het Agentschap Wegen en Verkeer. Eén van jouw taken bestaat erin om regelmatig langs alle snelwegen te rijden en ze controleren op mogelijke schade. Onderstaande graaf toont een dergelijk netwerk van wegen. De knopen zijn de verschillende steden, de bogen zijn de snelwegen die deze steden verbinden.
Veronderstel dat je in stad E woont, is het dan mogelijk om je rondrit te beginnen en te eindigen in E, waarbij je elk stukje autostrade precies één keer berijdt?
Geef zo'n rondrit.

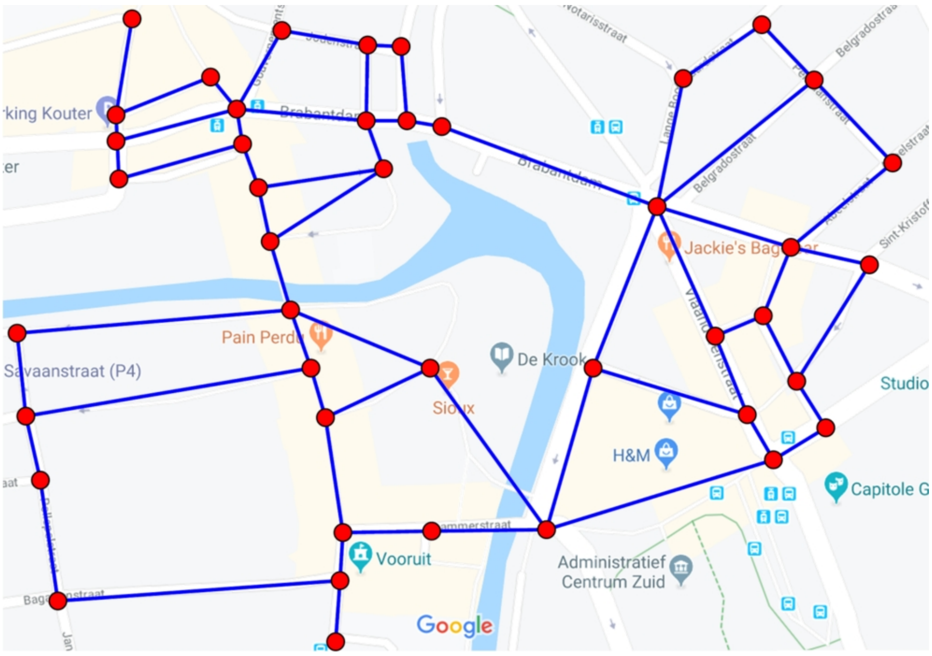


11. Duid een euleriaans circuit aan. Doe dit door de bogen opeenvolgend te nummeren.



Module 2

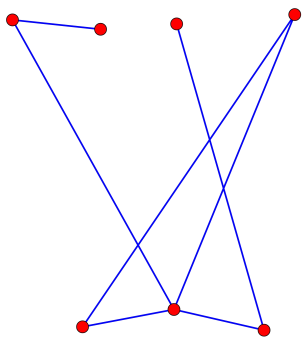
Grafen als modellering van de werkelijkheid



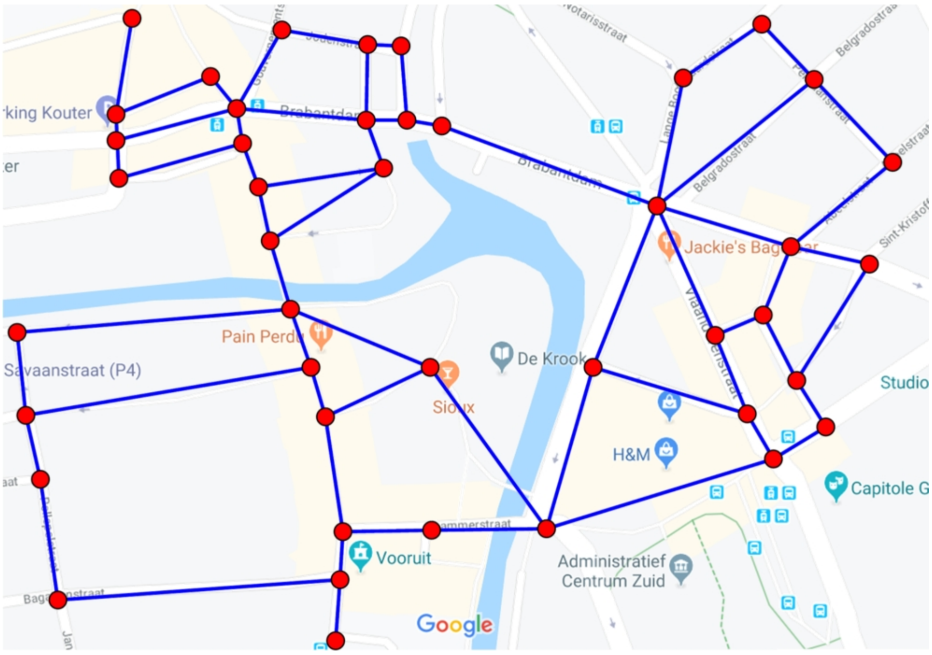
2.1 Abstractie van concrete situaties

De vorige module vertrok vanuit een concrete situatie – het bruggenprobleem van Königsberg – en abstraheerde dit naar een graaf. Dergelijke abstractie verwijdert onnodige randinformatie en beschrijft de kern van het vraagstuk. Zo kun je gemakkelijker over het probleem nadenken en eigenschappen over de graaf bewijzen.
In het vorig hoofdstuk stelden de knopen van een graaf de eilanden van Königsberg voor, de bogen stelden de bruggen voor. Besef dat grafen op diverse andere situaties kunnen worden toegepast. Zo kunnen knopen ook personen voorstellen op een feestje, waarbij twee knopen verbonden worden met een boog als die twee personen elkaar kennen.

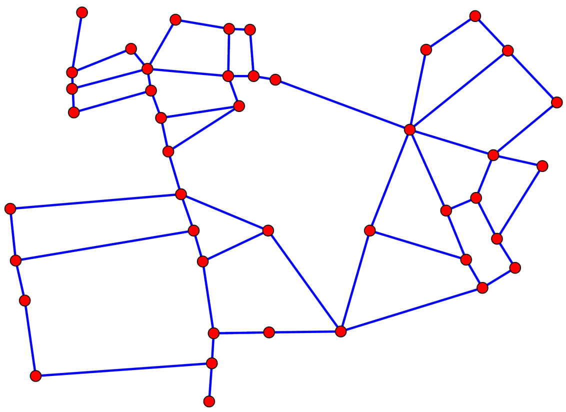
Vervolgens kunnen we alle 'decorerende' informatie weggommen, zodat we ons kunnen concentreren op de essentie van de situatie.



Anderzijds kan een stratenplan worden voorgesteld door een graaf door kruispunten te interpreteren als knopen en straten tussen twee kruispunten als bogen.



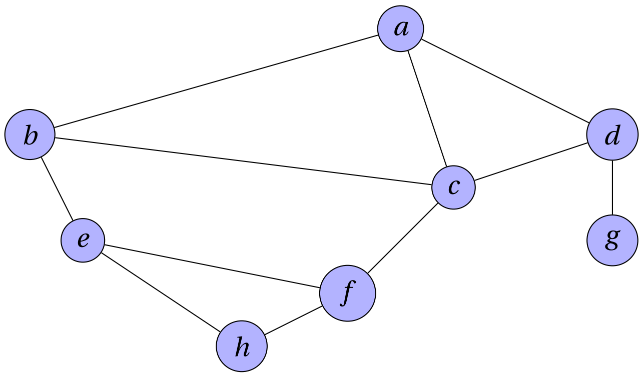
Straat- en plaatsnamen, rivieren... kunnen achterwege gelaten worden, zodat we slechts de graaf zelf overhouden.



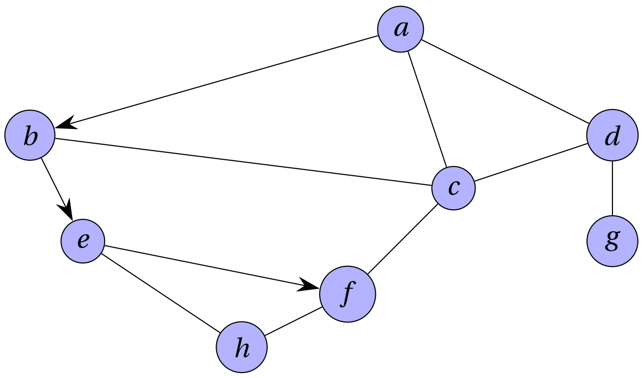
Besef dat de **interpretatie** van elk van de grafen verschilt. Als een knoop uit de eerste graaf graad 3 heeft, betekent dit dat de corresponderende persoon 3 anderen kent op het feest. Een knoop van graad 3 uit de tweede graaf komt echter overeen met een kruispunt dat drie straten verbindt. Omdat beide situaties kunnen worden **gemodelleerd** aan de hand van een graaf, kunnen we deze allebei tegelijkertijd bestuderen zonder ons te laten afleiden door wat elk van hun interpretaties precies is.

2.2 Gerichte grafen

In sommige situaties volstaat de eenvoud van een graaf niet om de kern van het probleem te beschrijven zonder essentiële informatie te verliezen. Eerder beschreven we een vriendschapsrelatie als symmetrisch ('als ik jou ken, ken jij mij'), maar dit is een veronderstelling en dus niet van toepassing in elke situatie. Om een ander voorbeeld te geven, stel dat we willen navigeren tussen verschillende plaatsen $\{a, b, c, \ldots, h\}$ in de stad. Zoals ons eerdere navigatievoorbeeld plaatsen we een boog tussen elke twee plaatsen die direct met elkaar verbonden zijn door een straat.



Als bovenstaande graaf een model is voor een navigatiemiddel voor auto's, dan is de kans groot dat sommige straten **eenrichtingsstraten** zijn. Het model moet hier uiteraard rekening mee houden als deze een routebeschrijving wilt maken. Dit valt echter gemakkelijk op te lossen als we bogen **gericht** mogen maken.

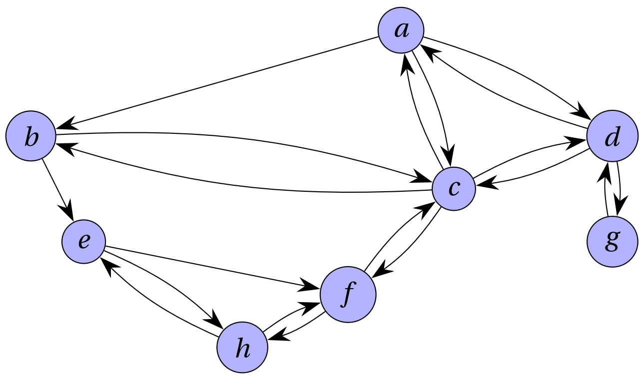


In bovenstaand voorbeeld is het niet meer mogelijk om rechtstreeks van knoop e naar knoop b te rijden; om b te bereiken moet je zeker knopen f en c passeren. Op deze manier kunnen sterke navigatietools zoals Google Maps éénrichtingsstraten aan het systeem toevoegen.

Definitie

Een **gerichte boog** van een graaf is een boog met een specifieke begin- en eindknoop. Een **gerichte graaf** is een graaf waarvan alle bogen gerichte bogen zijn.

Om bovenstaand voorbeeld te transformeren in een **gerichte** graaf, kunnen we elke ongerichte boog vervangen door twee gerichte bogen.



Merk op dat begrippen zoals wandeling, spoor, samenhangendheid... gemakkelijk te vertalen zijn naar gerichte grafen mits deze enkel een extra restrictie opleggen.

2.3 Graafkleuringen

2.3.1 Examens plannen

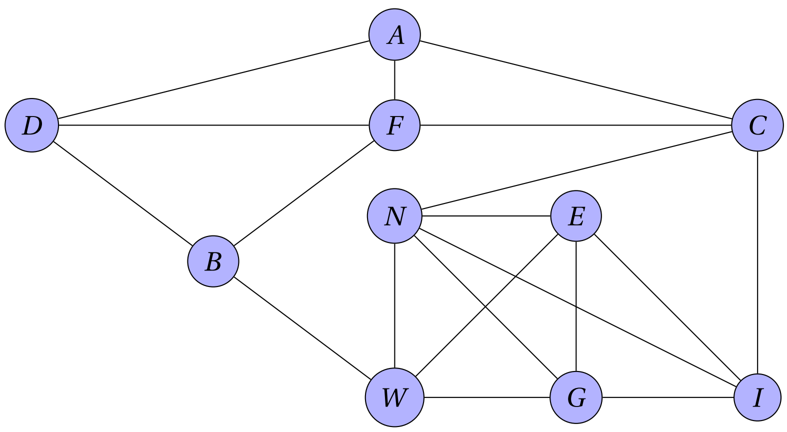
Tot nu toe hebben we enkel 'straight-forward' voorbeelden van modellen besproken. Bij alle grafen was de link tussen het model en de werkelijkheid haast tastbaar: straten en kruispunten van straatplan, feesters en hun onderlinge vriendschapsrelaties, eilanden en bruggen...
Een graaf kan echter ook een helder antwoord bieden op een complex administratief of logistiek probleem. Het model is correct, maar de link met de werkelijkheid is vaak niet zo eenvoudig als voorgaande voorbeelden. Beschouw de volgende probleemstelling.

Een aantal leerlingen van de school dienen deze zomer herexamens af te leggen. De volgende tabel geeft aan welke leerlingen welke herexamens moeten afleggen.

	Ayoub	Bo	Camille	Dagmar	Emma	Fahed	Gil	Hamza
Aardrijkskunde						X	X	
Biologie				X				X
Chemie			X				X	
Duits				X		X		
Engels	X				X			
Frans				X			X	
Geschiedenis	X	X						
Informatica	X		X					
Nederlands		X	X		X			
Wiskunde		X			X			X

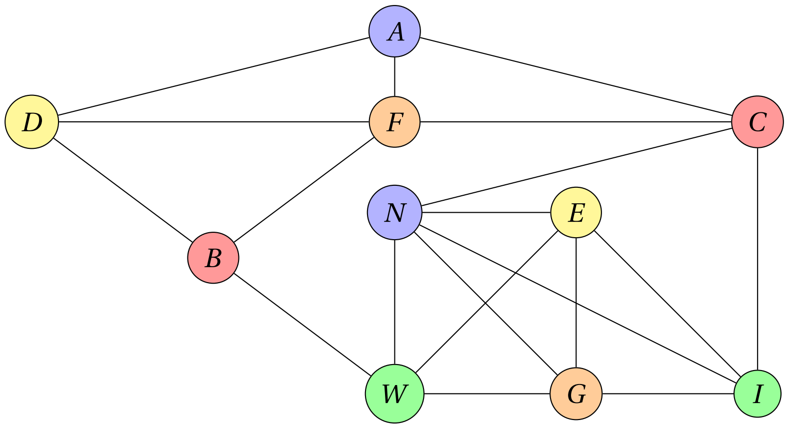
Er is echter een logistieke tegenslag: omdat de school aan het verbouwen is, blijft er slechts één week over wanneer de examen kunnen plaatsvinden, één examenmoment per schooldag. Op elk van deze 5 *examenmomenten kunnen er uiteraard wel meerdere examens tegelijk doorgaan.* ***Hoe kunnen deze examens ingepland worden zodat elke leerling kan deelnemen aan al zijn/haar/hun herexamens?***

Om bovenstaand probleem bevattelijker voor te stellen, gebruiken we een graaf. Twee examens mogen niet samenvallen als een leerling ze allebei moet afleggen. Dit kunnen we visualiseren aan de hand van een boog. Samengevat bestaat onze graaf uit 10 knopen, één voor elk examen (we labelen deze met de beginletters van elk vak), waarbij twee knopen verbonden zijn met elkaar als en slechts als een leerling ze allebei moet afleggen.



Om helder aan te geven welk examen op welke dag plaatsvindt, geven we elke dag (maandag tot vrijdag) een verschillend kleur en kleuren we een knoop in een bepaald kleur als het corresponderend examen plaatsvindt op de dag die met dat kleur gepaard gaat. We gebruiken bijvoorbeeld de kleuren maandag, dinsdag, woensdag, donderdag en vrijdag. Om te garanderen dat er voor geen enkele leerling overlap ontstaat, is het belangrijk dat naburige knopen

verschillende kleuren hebben. Gegeven deze spelregels kun je elke knoop van de graaf een kleur geven. Een voorbeeld van dergelijke **graafkleuring** is de volgende.



Uit bovenstaand schema halen we onmiddellijk:

- Maandag: Duits en Engels.
- Dinsdag: Informatica en Wiskunde.
- Woensdag: Biologie en Chemie.
- Donderdag: Aardrijkskunde en Nederlands.
- Vrijdag: Frans en Geschiedenis.

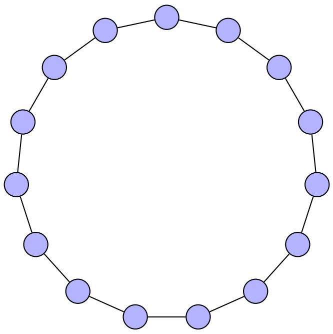
Onderzoeksopdracht 1
De leerlingen rebelleren! Ze willen op woensdag een rustdag om de zware examenweek te verlichten. Lukt het om de week te herplannen zodat alle examens op vier dagen vallen en elke leerling nog steeds kan deelnemen aan elk van zijn/haar/hun herexamens? Zouden de examens zelfs op drie dagen ingepland kunnen worden?

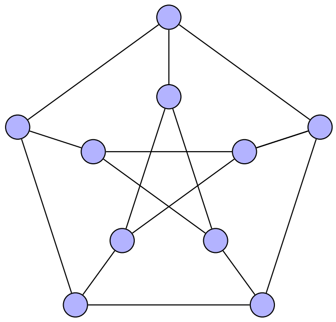
In bepaalde situaties is het belangrijk om te weten met hoe **weinig** kleuren je een graaf zoals bovenstaande kan inkleuren (zodat elke twee naburige knopen een verschillend kleur hebben).

Definities

Bij een **knopenkleuring** met k kleuren krijgt elke knoop een kleur (uit k keuzes) zodat naburige knopen een verschillende kleur hebben.
Het **chromatisch getal** $\chi(G)$ van een graaf G is gelijk aan het laagst aantal kleuren waarvoor er een knopenkleuring bestaat.

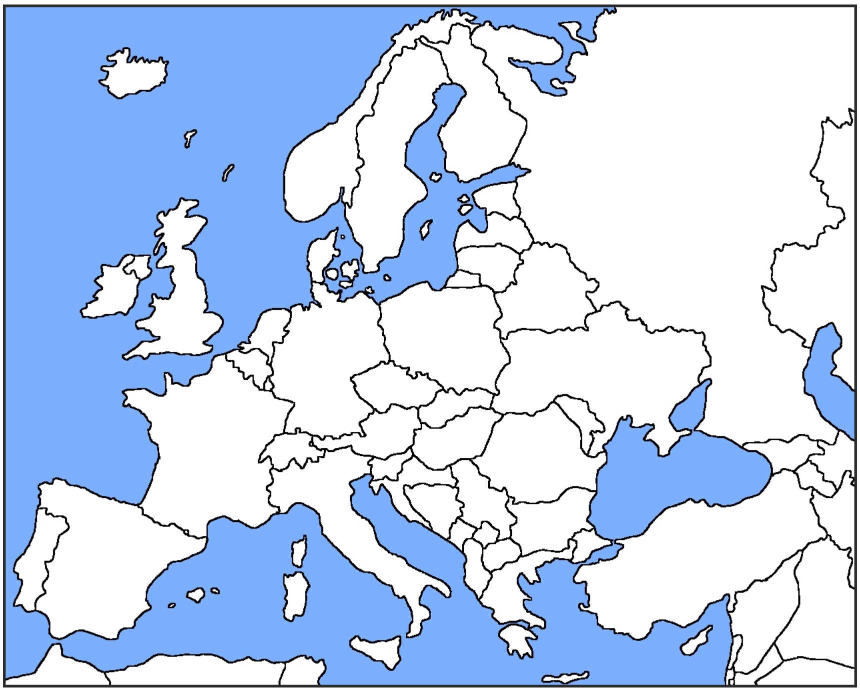
Opdracht
Bepaal de chromatische getallen van de volgende twee grafen.





2.3.2 Kaarten en kleuren

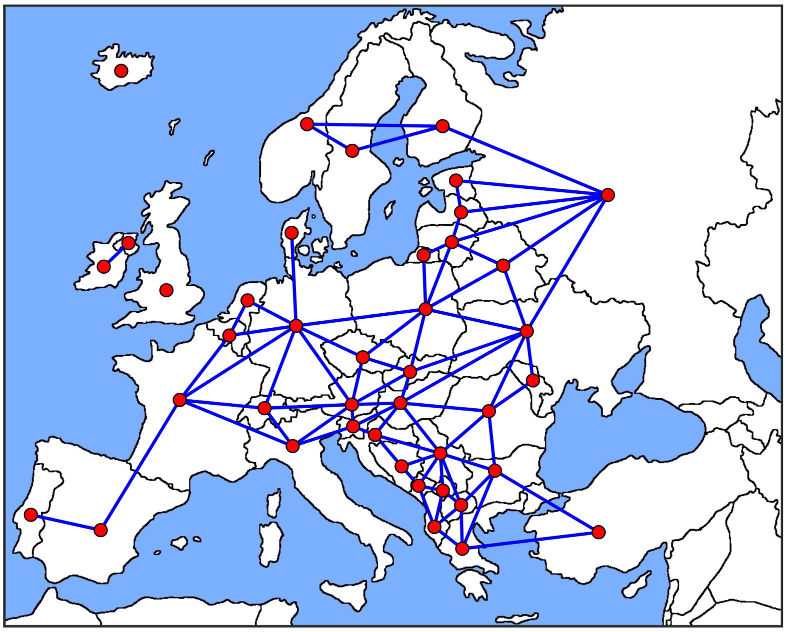
Stel, je wil graag een grote landkaart van Europa in je slaapkamer ophangen. Het internet biedt je er talloze aan en na wikken en wegen bestel je de volgende.



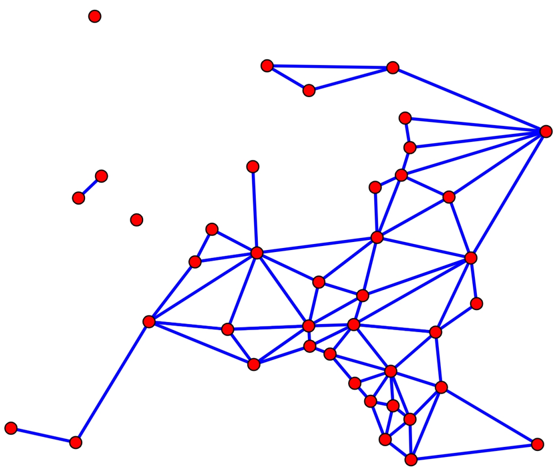
Je initiële vreugde wordt echter vervaagd door je verlangen naar meer kleur op je kamer. Je zou daarom graag elk land van je landkaart van een kleur voorzien. Om te vermijden dat twee naburige landen per ongeluk worden aanzien als één, hou je de volgende regel aan: **elke twee aangrenzende landen[1] hebben een verschillend kleur.**

[1] Hierbij maken we de afspraak dat twee landen pas *aangrenzend* zijn als ze een stuk grens (niet slechts een punt) gemeen hebben.

Dit vraagt naar abstractie naar de wereld der grafen. Als we elke land associëren met een knoop en een boog tussen twee knopen toevoegen als de bijhorende landen naburig zijn, bekomen we de volgende graaf.



Merk op dat we voor onszelf de situatie hebben vereenvoudigd: we beschouwen voornamelijk Europese landen, waarbij we relatief kleine landen negeren en naburigheid enkel erkennen als de landen er op de kaart visueel uitzien als buurlanden. Eens de graaf is geconstrueerd, vertaalt het probleem zich eenvoudigweg naar **het vinden van een knopenkleuring** binnen de volgende graaf.



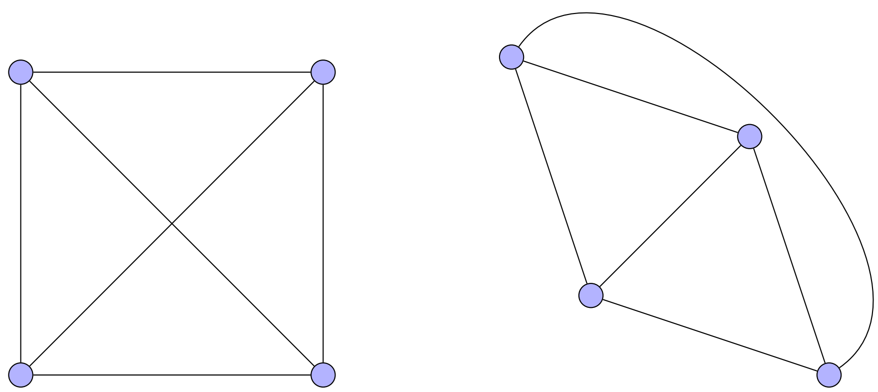
Er valt bovendien nog iets op aan een dergelijke graaf. We kunnen namelijk zo'n graaf **altijd** zodanig tekenen dat geen enkele twee bogen elkaar snijden (buiten in een knoop). Wetende dat dergelijke graaf afkomstig is van een landkaart, is deze eigenschap best vanzelfsprekend.

Definitie

Een graaf noemen we **planair** als ze visueel voorgesteld kan worden zonder dat bogen elkaar snijden (behalve in een knoop).[a]
Die voorstelling noemen we de **vlakke voorstelling** van de graaf.

[a]De bogen die je tekent hoeven niet recht te zijn. Wiskundig werd echter bewezen dat je elke planaire graaf zodanig kan tekenen dat alle bogen daadwerkelijk recht zijn!

Een graaf als abstractie van een landkaart is altijd een planaire graaf. Let wel op: bovenstaande definitie vertelt ons dat ook grafen waar bogen elkaar snijden evengoed planair kunnen zijn. Zo is de complete graaf [2] van orde 4 een planaire graaf, want het heeft een vlakke voorstelling, ondanks je initieel zou denken dat bogen sowieso snijden:



Onderzoeksopdracht 2
Voor planaire grafen geldt **de formule van Euler**. Deze formule geeft een verband weer tussen het totaal aantal knopen (= v), het totaal aantal bogen (= e) en het totaal aantal gebieden (= f) van een planaire graaf. Een **gebied** is een regio in de vlakke voorstelling van een graaf die volledig omringd is door bogen; we maken de afspraak dat de 'buitenomgeving' van een graaf ook één gebied vormt. Zo heeft de complete graaf van orde 4 precies $v = 4$ knopen, $e = 6$ bogen en $f = 4$ gebieden.
Teken enkele planaire grafen en ga telkens op zoek naar de bijhorende waardes voor v, e en f. Probeer uit je data een verband tussen v, e en f te vinden.

[2]Een graaf is compleet als elke 2 knopen door een boog met elkaar verbonden zijn.

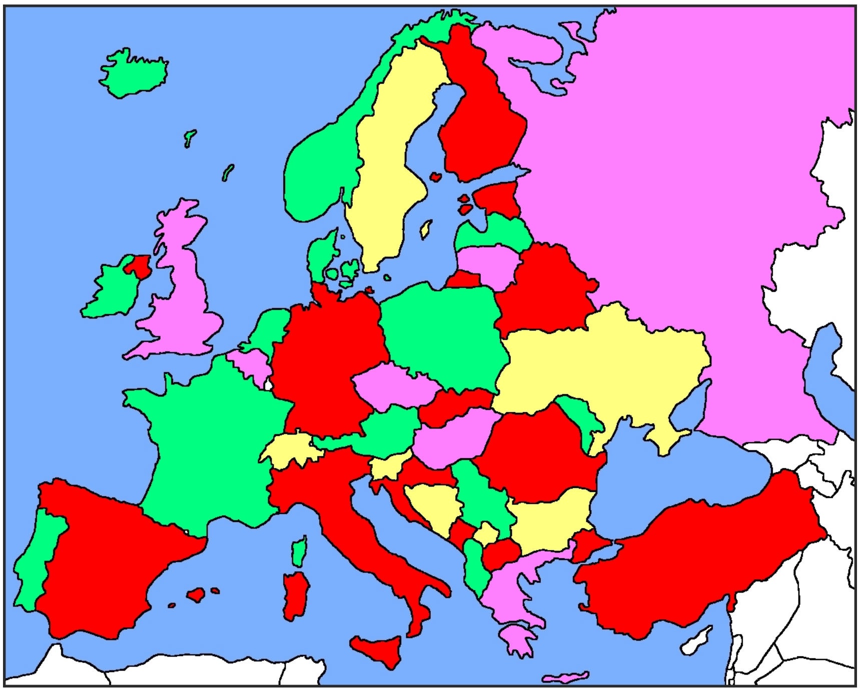
Om deze module af te sluiten kunnen we opnieuw de kleuring van de landkaart erbij nemen. Stel dat je maar vier verschillende kleurpotloden hebt om mee te kleuren. Wat is het minimum aantal kleuren die je nodig heeft om er zeker van te zijn dat er een juiste kleuring van de landkaart bestaat? Met andere woorden: **wat is het chromatisch getal van een planaire graaf?**

Wat blijkt, deze vraag bezorgde diverse wiskundigen vele slapeloze nachten. In 1852 vermoedde ene Francis Guthrie dat je eender welke landkaart kan inkleuren met ten hoogste vier kleuren. In 1879 juichte de wiskundige Alfred Kempe: hij publiceerde een bewijs van dit vermoeden. Elf jaar lang werd dit aangenomen als waar, tot Percy Heawood in 1890 een fout in het bewijs vond. Pas in 1976 werd door de wiskundigen Kenneth Appel en Wolfgang Haken een nieuw bewijs gevonden met behulp van een computer.

Vierkleurenstelling

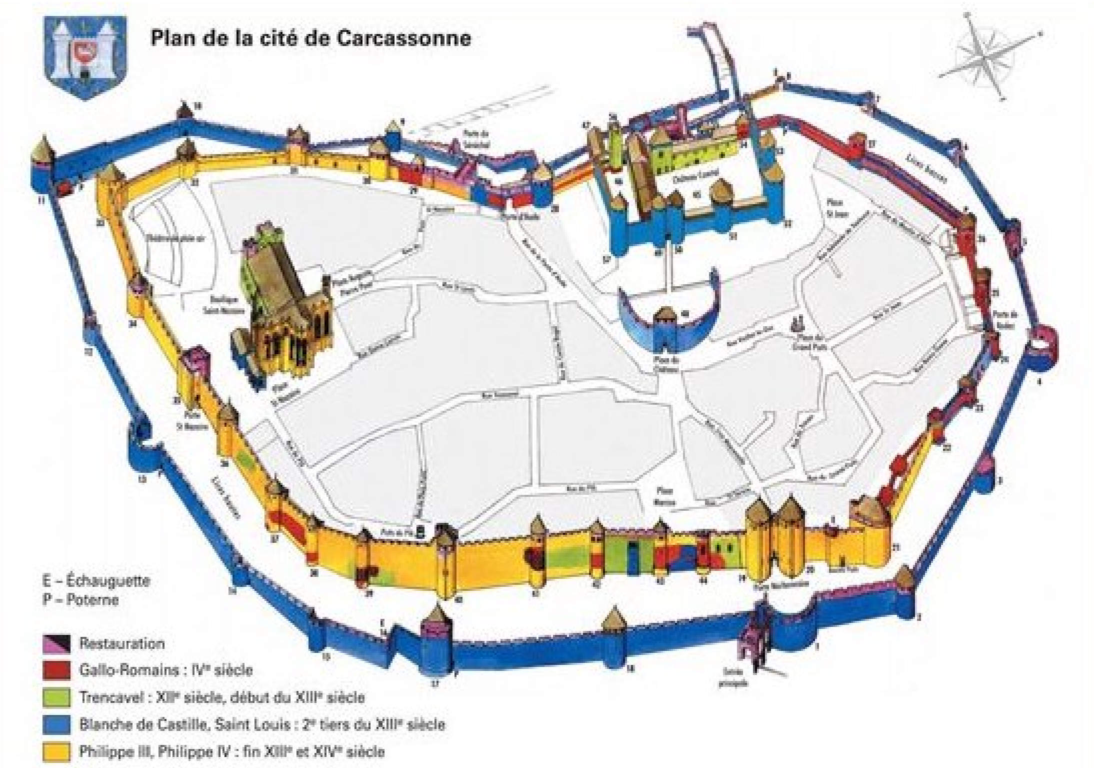
Als G een planaire graaf is, dan is $\chi(G) \leqslant 4$.

Elke landkaart is dus inkleurbaar met ten hoogste vier kleuren!



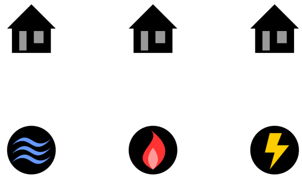
2.4 Oefeningen

1. Carcassonne is een vestingstad in Frankrijk. Het is na Parijs en de Mont-Saint-Michel de grootste toeristische trekpleister van Frankrijk. De bekendste bezienswaardigheid is de volledig gerestaureerde versterkte oude binnenstad, La Cité de Carcassonne. Deze Ville fortifiée historique de Carcassonne staat sinds 1997 op de werelderfgoedlijst van de UNESCO. Het is een uitzonderlijk voorbeeld van een middeleeuwse stad in Europa, die bijna volledig is bewaard, en de grootste behouden vesting is uit de middeleeuwen.



Stel de plattegrond van de ommuurde oude binnenstad voor d.m.v. een graaf.

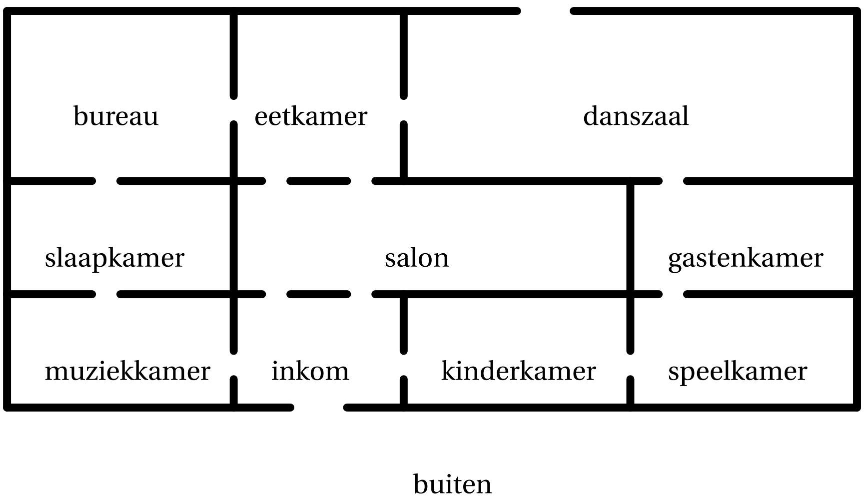
2. Het volgende probleem staat bekend als ‘three utilities problem’: kunnen de leidingen voor 3 nutsvoorzieningen naar 3 huizen zo worden gelegd dat geen van de leidingen elkaar kruisen?



Beantwoord volgende vragen.

   (a) Welke soort graaf voldoet aan de gestelde eis voor het leggen van de nutsvoorzieningen?
   (b) Probeer de nutsvoorzieningen met elk van de huizen te verbinden zodat de leidingen elkaar niet kruisen.
   (c) Kan je een dergelijke graaf tekenen? Verklaar je antwoord.
   (d) Bekijk: `https://www.youtube.com/watch?v=a8nOBXOBL6E`.

3. Gegeven de plattegrond van het huis waar Dr. Black is vermoord.



Inspecteur Pratt moet de moord op Dr. Black oplossen. Het lijk van Dr. Black is in het bureau teruggevonden. De buttler beweert dat hij iemand gezien heeft die in het bureau binnenging en via dezelfde deur het bureau weer verliet. De inspecteur ondervraagt Mrs. Peacock en die zegt: ‘Ik ben niet degene die Dr. Black gezien heeft, want ik ben door de voordeur (aan de inkom) binnengekomen en via de achterdeur (aan de danszaal) weer buiten gegaan en ik ben juist één keer door elke deur gegaan.’ Is iemand aan het liegen?

(a) Teken een multigraaf waarbij de kamers (en ook de buitenwereld) knopen zijn. De deuren vormen de bogen.

(b) Herformuleer de onderzoeksvraag m.b.v. grafen.

(c) Leg het verband met het probleem omtrent de zeven bruggen van Königsberg (zie vorige module) en beantwoord de vraag.

4. Een hogeschool biedt voor de aanvang van een nieuw academiejaar vakantiecursussen aan. In de laatste vakantieweek kan je inschrijven op dinsdag, woensdag, donderdag en vrijdag voor cursussen wiskunde, fysica, aardrijkskunde, geschiedenis, Engels, Frans en Duits. Elke cursus duurt een volle dag. Er zijn drie docenten: één geeft zowel wiskunde als fysica, een ander geeft geschiedenis en aardrijkskunde, en de derde geeft de moderne vreemde talen. Sommige ingeschreven deelnemers willen twee vakken volgen. Zo zijn er studenten die wiskunde willen combineren met Frans, of met aardrijkskunde, of met geschiedenis. Verder zijn er bij elke taal studenten die ook aardrijkskunde of geschiedenis willen volgen. Tot slot zijn er nog studenten die naast fysica ook Engels willen leren. Andere combinaties komen niet voor.
Lukt het de zeven vakken in vier dagen te geven, zo dat alle studenten die twee vakken willen volgen dat ook kunnen?

   (a) Stel de situatie zodanig voor in een graaf, dat je op zoek kan gaan naar een antwoord op de vraag.

   (b) Beantwoord de vraag.

5. **Verdiepingsvraag: boogkleuringen**
We bekijken nog een ander roosterprobleem dat je met kleuringen kan oplossen. Er zijn zes leerkrachten: A, B, C, D, E en F, zes klassen: I, II, III, IV, V en VI en drie lesuren: 1, 2 en 3. In onderstaand schema geven de witte hokjes aan wie aan welke klassen moet lesgeven, bijvoorbeeld klas I krijgt les van docent C, D en F. Door in elk leeg hokje één van de cijfers 1, 2 of 3 in te vullen ontstaat een lessenrooster. Het cijfer 2 in hokje $C - V$ zou betekenen dat docent C in lesuur 2 lesgeeft aan klas V.

	A	B	C	D	E	F
I						
II						
III						
IV						
V						
VI						

Er gelden bovendien twee voorwaarden voor dit rooster.

- Een leerkracht kan maar aan één klas tegelijk lesgeven.
- Een klas kan maar van één leerkracht tegelijk les krijgen.

Beantwoord volgende vragen.

(a) Wat betekenen de voorwaarden voor de kolommen en rijen van het schema?

(b) Probeer m.b.v. de tabel een passend lessenrooster te maken. Hoe vlot lukt dit?

(c) Vertaal het roosterprobleem naar een graaf waarbij je zowel de leerkrachten als de vakken voorstelt door een knoop. Plaats de knopen voor de leerkrachten naast elkaar op een rij. Doe hetzelfde voor de knopen die vakken voorstellen, maar plaats deze op een rij eronder.
Een dergelijke graaf wordt een **tweedelingsgraaf** of **bipartiete graaf** genoemd, omdat de knopen zijn verdeeld in twee groepen zodanig dat elke boog van de graaf een knoop van de ene groep verbindt met een knoop van de andere groep. Er mogen dus geen bogen zijn tussen knopen van eenzelfde groep.

(d) Hoe kan je nu een lessenrooster maken m.b.v. deze graaf? Om het probleem te kunnen oplossen, welke betekenis geef je aan een boog? Hoe maak je onderscheid tussen de lesuren 1, 2 en 3?

(e) Probeer een lessenrooster te vinden.
Opmerking: de stelling van König zegt dat je in een bipartiete graaf met maximale graad k de bogen kan kleuren met k kleuren zodanig dat gelijk gekleurde bogen geen knoop gemeenschappelijk hebben. Aangezien in deze graaf de graad van elke knoop 3 is, zijn we ervan verzekerd dat er een lessenrooster kan worden gevonden. Er bestaan algoritmes om dergelijke problemen op te lossen.

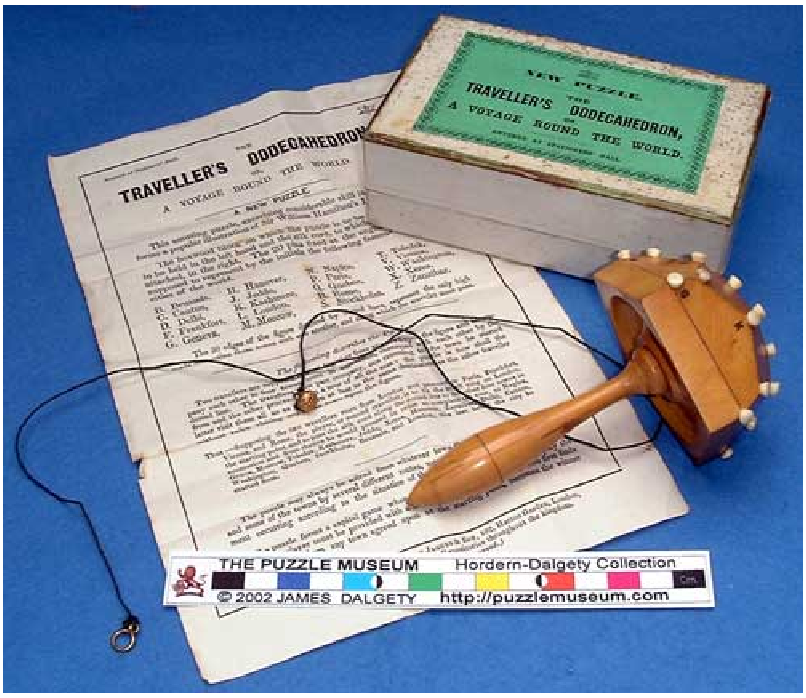
6. Bepaal het chromatisch getal van een bipartiete graaf.

Module 3

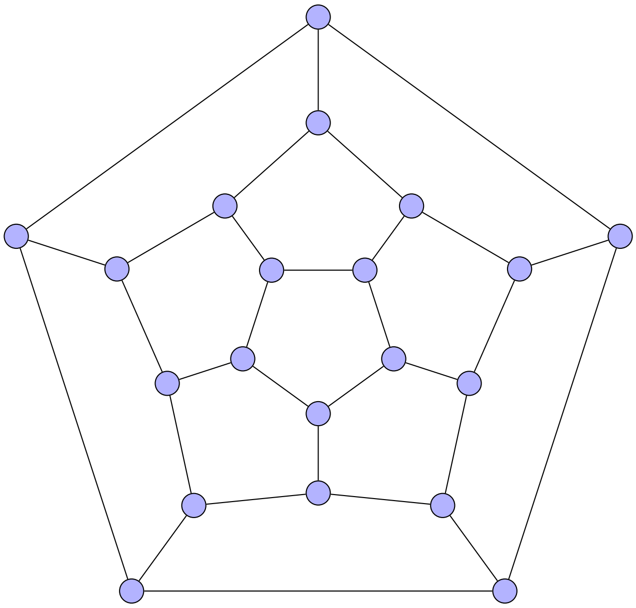
Handelsreizigers

3.1 Hamiltoniaanse grafen

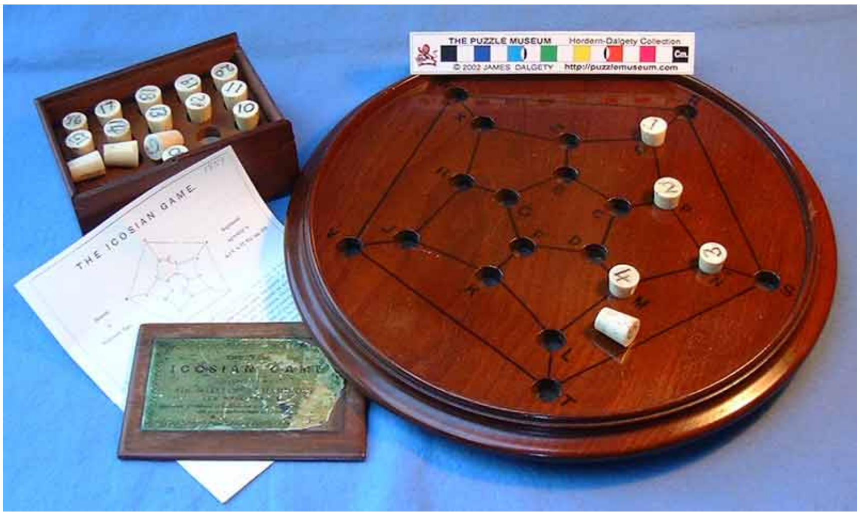
Sir William Rowan Hamilton was een Ierse wiskundige uit de 19de eeuw. Hij was professor astronomie aan de *Trinity College* te Dublin, en zelfs 'Royaal Astronoom van Ierland'. Ondanks diverse academische verwezelijkingen binnen de fysica, is zijn naam binnen de wiskunde vooral blijven hangen vanwege... een spelletje.



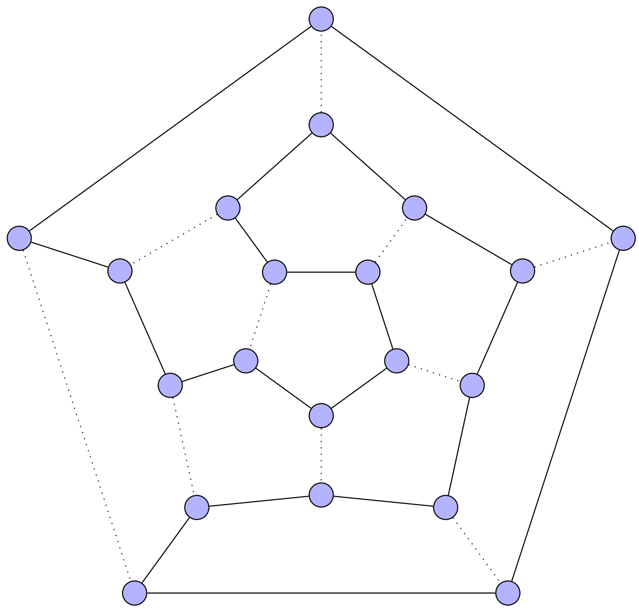
Het spelletje heette '*Traveller's dodecahedron: A voyage around the world*' (dodecaëder van de reiziger: een reis rond de wereld) en bestond uit een houten object in de vorm van een paddenstoel, met daarop 20 spelden. Elke speld stelt een wereldstad voor, die jij als reiziger wil bezoeken. Aan de hand van een zijden draad moest de speler erin slagen, vertrekkend vanuit één speld, een pad te vormen langs alle 20 spelden – zónder tweemaal langs dezelfde speld te passeren – om opnieuw uit te komen bij de speld waar men vertrok.
Zoals de naam van het spel verraadt, is de vorm van de paddenstoel afgeleid van een dodecaeder, met op elk hoekpunt een speld. Als we elk hoekpunt van een dodecaëder associëren met een knoop en elke ribbe met een boog, en vervolgens deze graaf projecteren op een vlak, dan bekomen we volgende planaire graaf.



Bovenstaande vorm was zelfs de oorspronkelijke vorm van het spel.



Deze variant sloeg echter niet aan bij haar kopers, dus bedacht Hamilton de paddenstoelvariant. De reden waarom bovenstaande 'vlakke' variant niet aansloeg, was omdat het simpelweg te gemakkelijk was. Eens je de puzzel kan voorstellen in het vlak, kan je relatief snel een cyclische route bedenken die geen enkele knoop meerdere keren bezoekt:



Dergelijke route zal echter niet in elke graaf aanwezig is (denk maar aan een graaf met een knoop van graad 1). Vandaar komt de naam **hamiltongraaf**, speciaal voor grafen waar zo'n route wél mogelijk is. We herformuleren dit soort problemen waarbij elke knoop precies eenmaal moet worden bereikt m.b.v. grafen na het invoeren van onderstaande definities.

Definities

Een **pad** is een wandeling waarbij geen knopen meer dan eenmaal worden doorlopen.

Een **hamiltoniaans pad** is een pad dat alle knopen doorloopt.

Een gesloten pad of **cykel** is een pad dat vertrekt en eindigt in dezelfde knoop.

Een **hamiltoniaanse cykel** is een cykel dat alle knopen doorloopt.

Een **hamiltoniaanse graaf** of **hamiltongraaf** is een graaf die een hamiltoniaanse cykel bevat.

Hamiltoniaanse grafen zijn een typevoorbeeld waarbij een concreet planningsprobleem (of in dit geval: een spelletje) geleid heeft tot de definitie van een abstracte, graaftheoretische eigenschap.

Aan de hand van deze definities kunnen we het bovenstaande spel herleiden tot de vraag *is de corresponderende graaf een hamiltoniaanse graaf?*

Ook in het wiskundig onderzoek duiken hamiltoniaanse grafen af en toe op. Er bestaat zelfs een *voldoende voorwaarde* opdat een graaf hamiltoniaans is:

Stelling: Dirac

Zij G een graaf met $n \geqslant 3$ knopen. Als de graad van elke knoop groter dan of gelijk is aan $\frac{n}{2}$, dan is G hamiltoniaans.

Merk op dat bovenstaande voorwaarde geen *nodige* voorwaarde is, gezien de grafen die we hier als voorbeeld naar voren schoven telkens niet voldeden aan de voorwaarde op de graad van hun knopen, maar toch hamiltoniaans zijn.

3.2 Lengte van een pad

Definities

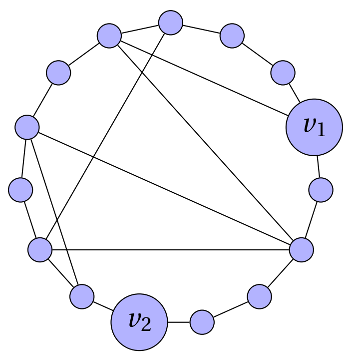
De **lengte** van een pad is het aantal bogen in dit pad.

Een **kortste pad** tussen knopen v_1 en v_2 is een pad met v_1 en v_2 als respectievelijke begin- en eindknoop dat bestaat uit een minimaal aantal bogen.

De **afstand** tussen knopen v_1 en v_2 is de lengte van een kortste pad tussen v_1 en v_2.

Opmerking: besef dat tussen twee knopen meerdere kortste paden kunnen bestaan.

Opdracht
Hoeveel kortste paden tel je tussen knopen v_1 en v_2 in volgende graaf?
Wat is de afstand tussen deze twee knopen?



3.3 Gewogen grafen

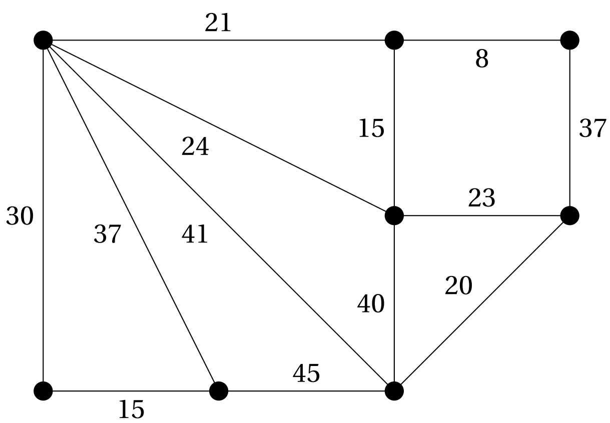
Men kan met 'kortste pad' verscheidene zaken bedoelen. Een afstandsgewijs kortste route van A naar B is niet per se de kortste qua tijd. Bovendien verschillen bepaalde routes sterk wat hun toegankelijkheid betreft (denk maar aan autosnelwegen versus hobbelige landweggetjes).
Om hiermee rekening te houden in hun abstracte voorstelling als grafen, introduceren we **gewogen grafen**. Hierbij kennen we een zeker **gewicht** toe aan elke boog. In de context van routeplanning kan dergelijk gewicht bijvoorbeeld de lengte of toegankelijkheid van de bijhorende straat aanduiden.

Definities

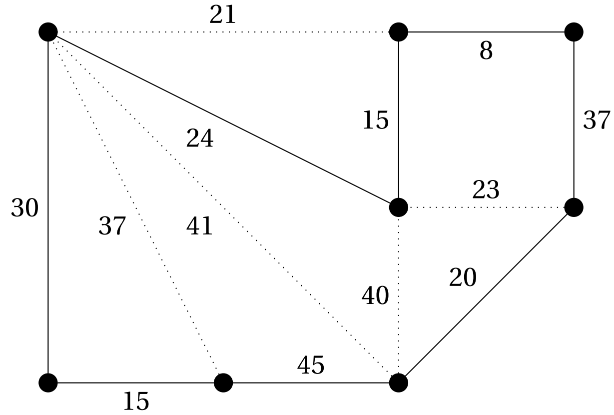
Een **gewogen graaf** is een graaf waarbij aan elke boog b een getal $g(b)$ gehecht is, dat het **gewicht** van de boog genoemd wordt.
Het **gewicht** $g(G)$ **van de graaf** G is de som van de gewichten van alle bogen van G.

Een **kortste pad** tussen knopen v_1 en v_2 in een gewogen graaf is een pad waarvan de som van de gewichten van de bogen minimaal is.

Voorbeeld
Onderstaande graaf stelt verschillende toeristische trekpleisters (knopen) voor in de regio waar je op vakantie bent. De knopen zijn verbonden met elkaar als er een rechtstreekse busverbinding bestaat. Het gewicht die met elke boog gepaard is stelt de afstand (in km) van de betreffende busverbinding voor.



Stel nu dat je op rondreis bent, en liefst elke bezienswaardigheid bezoekt. Omdat je geen nodeloze kilometers wil afleggen, wil je de afstand minimaliseren. Om je route te bepalen, moet je dus op zoek gaan naar **een hamiltoniaanse cykel met een kleinst mogelijk totaal gewicht**.



Hier is het totale gewicht van de cykel 194, oftewel 194 km. Je kan controleren dat dit daadwerkelijk de hamiltoniaanse cykel is van kleinste gewicht.

Bij een kleine graaf – zoals de bovenstaande – is de zoektocht naar dergelijke cykel relatief gemakkelijk. Vanaf je graaf groot wordt, echter, is dat niet zo – zelfs niet voor supercomputers! Dat is het idee achter het **handelsreizigersprobleem**.

3.4 Handelsreizigers

Handelsreizigersprobleem

Een handelsreiziger bezoekt een aantal mogelijke klanten, keert 's avonds terug waar hij 's morgens vertrok en wil hierbij een zo klein mogelijke afstand afleggen.

Om deze situatie te modelleren, interpreteert men best de klanten als knopen. Elke twee klanten worden verbonden door een boog (het is met andere woorden een complete graaf) en het bijhorende gewicht stelt de reisafstand voor tussen die twee klanten. Bij deze graaf vertaalt het probleem zicht tot: **vind de hamiltoniaanse cykel van kleinste gewicht**.

Het grote obstakel bij dit probleem is om net *de kleinste* van al die hamiltoniaanse cykels te vinden – en dat zijn er veel! Bij een complete graaf met 20 klanten/knopen bestaan er meer dan 60 biljard (!) mogelijke routes. Zelfs de krachtigste supercomputers zal even zuchten bij het overlopen van al deze cykels.

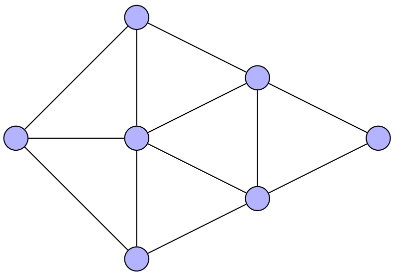
Misschien denk je nu: 'maar hé, waarschijnlijk bestaan er wel slimmere en efficiëntere methodes om de juiste cykel te bepalen!'.
Wel, dan heb je deels gelijk. Er bestaan wel degelijk een aantal betere algoritmes dan de naïeve methode om alle mogelijkheden te overlopen. De looptijd van deze efficiëntere algoritmes is echter niet *significant* lager. Formeler uitgedrukt: de looptijd van elk algoritme stijgt exponentieel snel vanaf de graaf in aantal knopen groter wordt.

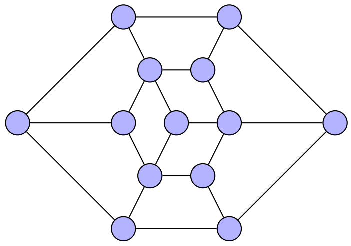
Het gekke is dat zowat alle 'exponentieel zware' problemen te vertalen zijn naar dit handelsreizigersprobleem. Als er dus iemand erin slaagt om een algoritme te bedenken met een *significant lagere* looptijd, dan kan dit algoritme tevens worden toegepast op duizenden andere problemen uit de informatica. Bovendien wordt die persoon er een miljoen US dollar rijker door. Op het vinden van dergelijk algoritme staat immers een geldprijs vast, want het is één van de zeven **millennium problemen**[1].

[1]https://www.claymath.org/millennium-problems/p-vs-np-problem.

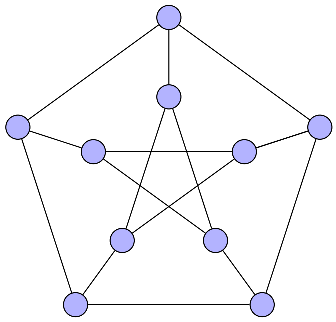
3.5 Oefeningen

1. Welke van de volgende grafen zijn hamiltoniaans? Waarom (niet)?

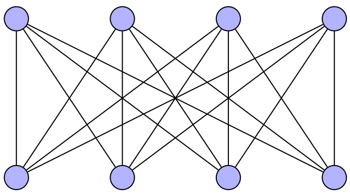




2. Is de petersengraaf een hamiltongraaf?

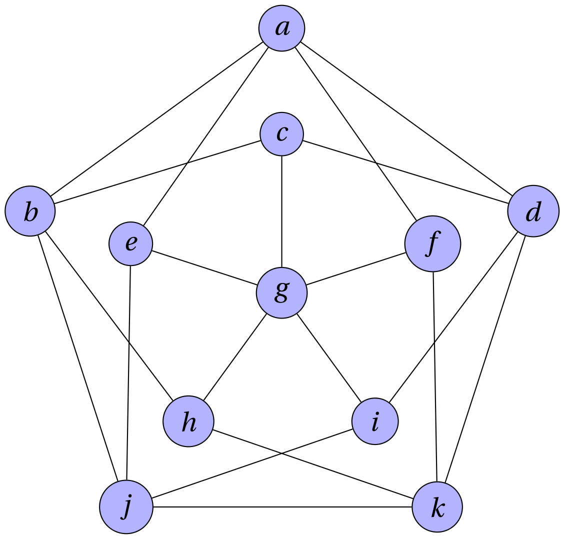


3. Is onderstaande bipartiete graaf een hamiltongraaf?



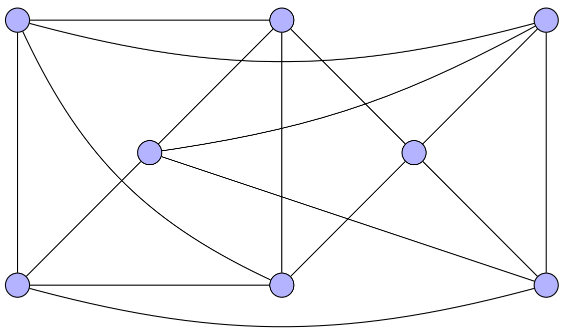
4. Op grondgebied Gent liggen er 6 treinstations voor reizigers: Gent-Sint-Pieters, Gent-Dampoort, Drongen, Merelbeke, Gentbrugge en Wondelgem.
   (a) Maak een gewogen graaf voor deze 6 stations.
   (b) Welke getallen kies je om als gewicht aan de bogen toe te kennen? Zoek de nodige gegevens op.
   (c) Bepaal het gewicht van deze graaf.

5. Beschouw onderstaande graaf.



   (a) Bepaal de lengte van het pad $P = (a, b, c, g, i, j)$.
   (b) Bepaal de afstand tussen de knopen a en i.
   (c) Is bovenstaande graaf een hamiltongraaf?

6. Maak gebruik van de stelling van Dirac om te bewijzen dat volgende graaf hamiltoniaans is. Zoek vervolgens een hamiltoniaanse cykel.



7. Een **paardenrondgang** is een route bestaande uit paardensprongen over een bord opgedeeld in vakjes zoals een schaakbord, waarbij elk vakje precies eenmaal wordt bezocht. Deze route kan op verschillende manieren beginnen en eindigen:

   - beginnen en eindigen op dezelfde plek (ook wel een gesloten rondgang genoemd),
   - beginnen en eindigen op twee vakjes direct naast elkaar,
   - beginnen en eindigen op twee willekeurige verschillende vakjes.

De oudst bekende vastgelegde paardenrondgang is een oplossing (1733) van de Franse wiskundige Abraham de Moivre. Leonhard Euler bedacht een rondgang die eerst de ene helft van het schaakbord (8×8 vakjes) afwerkte om vervolgens de tweede helft op te vullen. Eén van de meest briljante oplossingen is eveneens van Leonhard Euler. In deze oplossing presenteert hij een opgevuld schaakbord met een rondgang waarvan de op volgorde genummerde sprongen op het bord ook een (half)magisch vierkant [2] voorstellen met de magische som 260. Halfmagisch omdat de diagonalen niet de som van 260 opleveren, maar daar staat tegenover dat de vier kwadranten van het vierkant ook halfmagisch zijn. Het vermoeden bestaat dat er geen perfect magisch vierkant bestaat dat met de paardenrondgang gevuld kan worden.

Eulers briljante oplossing: een halfmagisch vierkant

8	01	48	31	50	33	16	63	18
7	30	51	46	03	62	19	14	35
6	47	02	49	32	15	34	17	64
5	52	29	04	45	20	61	36	13
4	05	44	25	56	09	40	21	60
3	28	53	08	41	24	57	12	37
2	43	06	55	26	39	10	59	22
1	54	27	42	07	58	23	38	11
	a	b	c	d	e	f	g	h

zie ook animatie op `https://nl.wikipedia.org/wiki/Paardenrondgang`

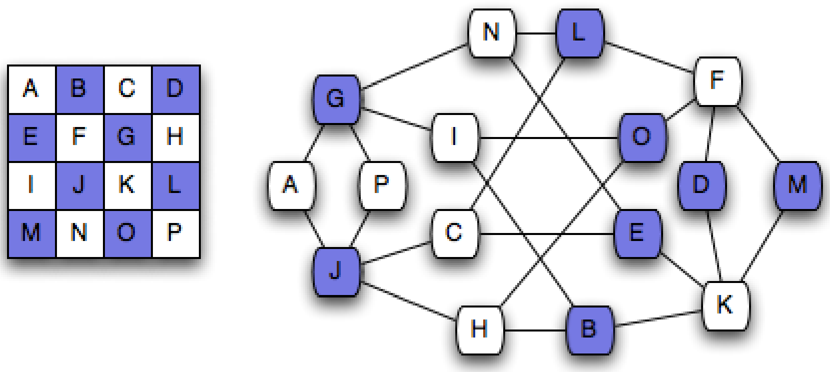
(a) Kan je een paardenrondgang vinden op een 3×3 schaakbord?

(b) Vertaal het probleem van een (gesloten) paardenrondgang naar een formulering met grafen.
Wat stemt overeen met knopen en bogen? Welke soort graaf komt overeen met een (gesloten) paardenrondgang?

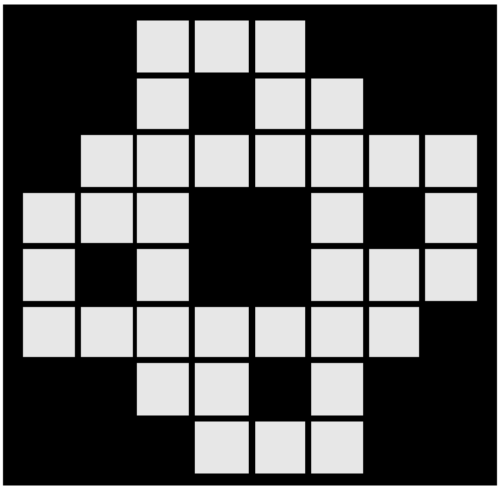
(c) Teken de bijhorende graaf bij een 3×3 schaakbord.

(d) Hieronder zie je de graaf die paardensprongen op een 4×4 schaakbord voorstelt. Toon m.b.v. deze graaf aan dat er op een 4×4 schaakbord geen paardenrondgang mogelijk is.

[2] Een magisch vierkant is een vierkant waarvan de som van de getallen in elke horizontale en verticale rij en in elke diagonaal gelijk is.



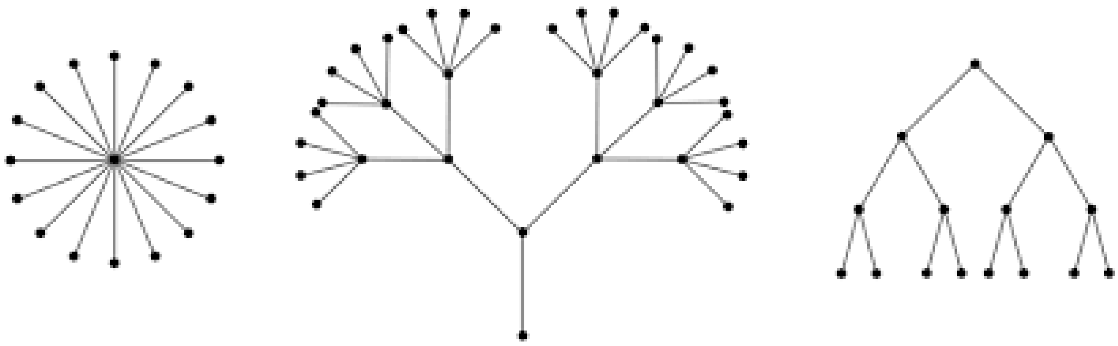
(e) Kan je een gesloten paardenrondgang vinden op onderstaand bord?



(f) Wie meer wil, kan talloze paardenrondgangpuzzels vinden op `https://www.archimedes-lab.org/knight_tour.html`.

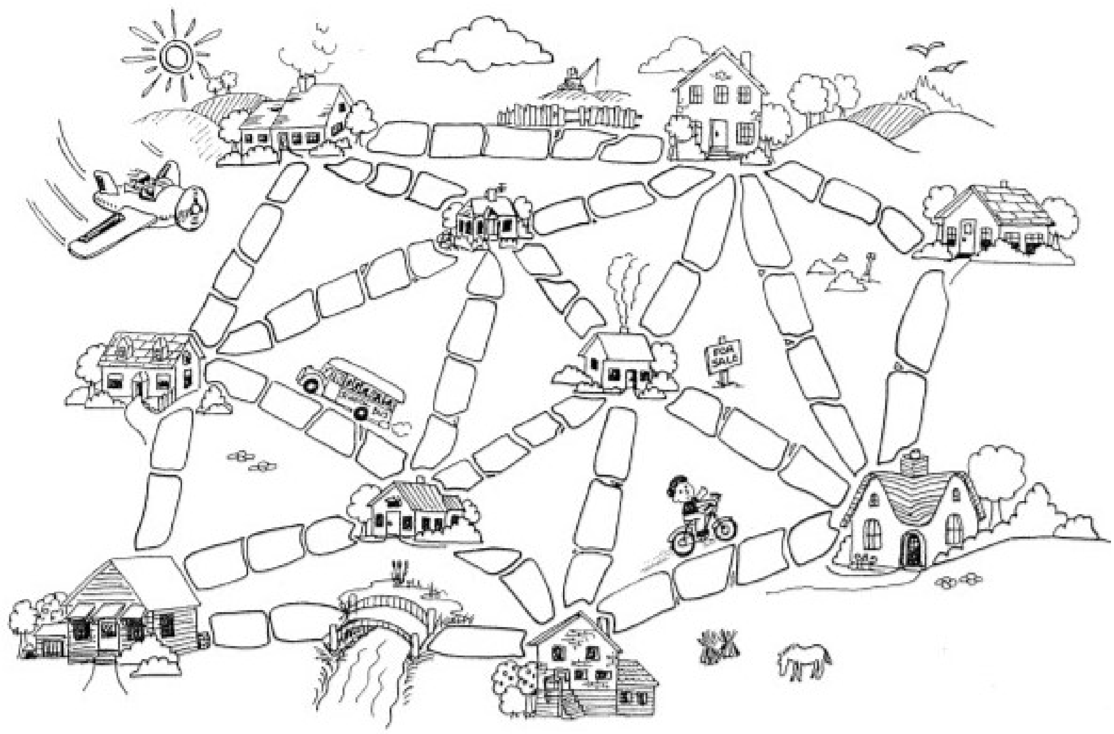
Module 4

Bomen

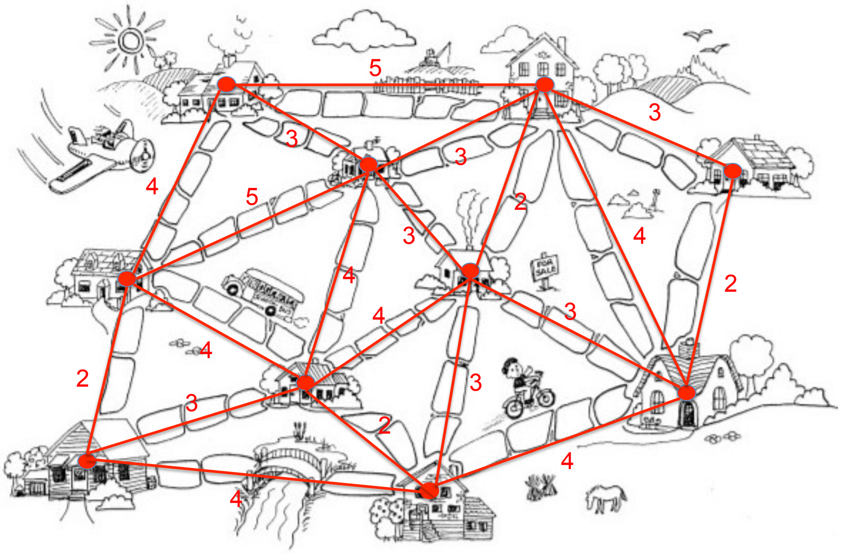


4.1 Inleidend voorbeeld: op zoek naar het goedkoopste netwerk

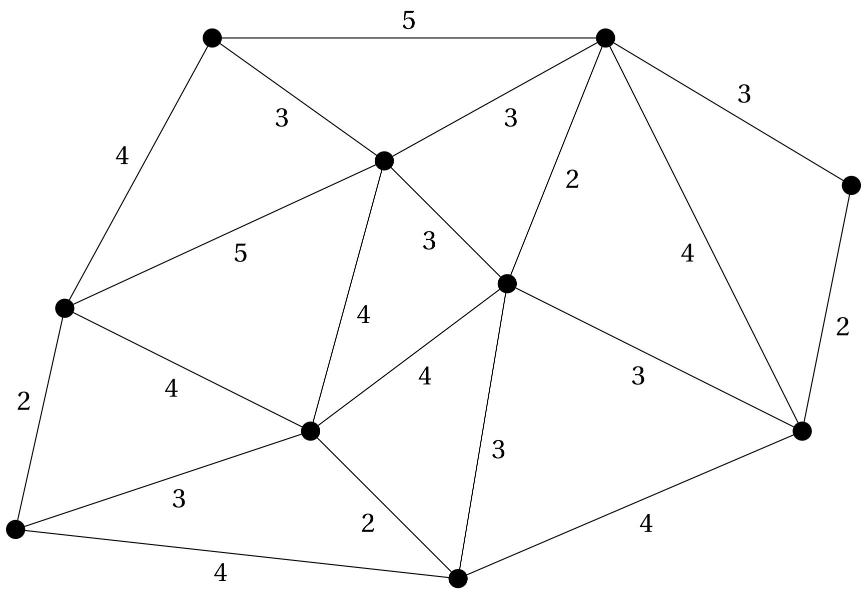
Een telecommunicatiebedrijf dient kabels aan te leggen in een vakantieoord (hieronder afgebeeld). Het bedrijf wil zo weinig mogelijk geld uitgeven, maar elke twee huizen moeten met elkaar verbonden zijn door het netwerk. Het aantal steentjes tussen elke twee huizen is evenredig met de prijs van een kabel tussen deze twee huizen. Hoe moet het bedrijf zijn kabels leggen?



We stellen bovenstaand netwerk voor als een graaf G: elke knoop stelt een huis voor, en twee huizen zijn verbonden door een boog als er een stenen pad de twee huizen verbindt. We noteren bij elke boog een getal dat het aantal stenen weergeeft tussen de huizen. Ook het bruggetje telt als steen.



Zoals steeds laten we de overtollige informatie weg en bekomen we onderstaande graaf.



Doordat er aan de bogen nu getallen (gewichten genoemd) zijn toegekend, noemen we een dergelijke graaf een gewogen graaf (zie ook module 3).

4.2 Herformulering van het probleem ‘op zoek naar het goedkoopste netwerk’

Om het probleem te herformuleren m.b.v. grafen, herhalen we de definities voor een gewogen graaf en het gewicht van een graaf. We voeren bovendien enkele nieuwe definities in.

Definities

Een **gewogen graaf** is een graaf waarbij aan elke boog b een getal $g(b)$ gehecht is, dat het **gewicht** van de boog genoemd wordt.
Het **gewicht** $g(G)$ **van de graaf** G is de som van de gewichten van alle bogen van G.

Een graaf $G' = (V', E')$ is een **deelgraaf** van een graaf $G = (V, E)$ als $V' \subseteq V$ en $E' \subseteq E$, met de afspraak dat E' enkel bestaat uit bogen die twee knopen uit V' verbinden.

We zijn dus op zoek naar een samenhangende graaf G‘ met een minimaal gewicht dat alle knopen van G omvat.
Als er cykels in het netwerk aanwezig zijn, dan zijn er zeker overbodige verbindingen gemaakt. Een dergelijke graaf wordt een boom genoemd. Aangezien deze boom alle knopen van G moet bevatten, spreken we van een opspannende boom.
We zoeken dus een opspannende boom met minimaal gewicht, dergelijke deelgraaf G' van G noemen we een minimaal opspannende boom van G.

Definities

Een **boom** is een samenhangende graaf die geen cykels omvat.
Een **opspannende boom** van G is een deelgraaf G' dat alle knopen van G bevat en een boom is, dus geen cykels bevat.
Een **minimaal opspannende boom** G' is een opspannende boom van een gewogen graaf G met minimaal gewicht.

Een criterium (alternatief voor de definitie) opdat een samenhangende graaf een boom is wordt gegegeven door volgende stelling.

Stelling

Een samenhangende graaf G met n knopen is een boom als en slechts als G $n-1$ bogen heeft.

Opdracht: herformuleer de probleemstelling m.b.v. bovenstaande definities.

Probleemstelling

Gegeven een gewogen graaf, hoe vinden we de minimaal opspannende boom?

4.3 Algoritme van Kruskal

We bekijken een methode, die gegarandeerd werkt, om in een gewogen graaf een minimale opspannende boom te vinden. Zo'n stapsgewijze methode wordt een **algoritme** genoemd. Eén van die methodes is het **algoritme van Kruskal**.
Typisch start een algoritme met een bepaalde beginstap, en dan een stap die je moet herhalen tot het gewenste resultaat bekomen wordt. Met welke graaf we ook starten, het toepassen van deze stappen zal ons een mimimale opspannende boom opleveren.
Het principe van het algoritme van Kruskal, genoemd naar zijn bedenker J.B. Kruskal (1956), is het uitvoeren van de volgende stap op een gegeven gewogen graaf G: kies uit de bogen van G die nog niet werden gekozen, de boog met het kleinste gewicht die geen cykel vormt met de bogen die reeds werden gekozen.
Als er geen kandidaten meer overblijven vormen de geselecteerde bogen de minimaal opspannende boom.

Meer specifiek ga je als volgt te werk.

Algoritme van Kruskal

Ga uit van een samenhangende gewogen graaf G met minstens 3 knopen en een graaf H met dezelfde knopen als G, maar nog geen bogen.

1. **Begin**: voeg een boog van G met het laagste gewicht toe aan H.
2. **Herhaal** volgende stap: kies een boog van G die nog niet tot H behoort zodat
   - de graaf die uit H ontstaat door de boog aan H toe te voegen geen cykel bevat en
   - het gewicht van die boog zo laag mogelijk is.
3. **Einde**: stop op het moment dat er bij stap 2 geen nieuwe boog gekozen kan worden. H is dan de minimaal opspannende boom van G.

Stap 3 zorgt ervoor dat het algoritme op het juiste moment stopt. Zonder deze stap zou je te vroeg kunnen stoppen. In stap 2 gebeurt het eigenlijke werk: hier bouw je stap voor stap een deelgraaf H van G op die aan het eind een minimaal opspannende boom van G oplevert. Het is belangrijk dat je in stap 2 steeds controleert dat er geen cykels ontstaan in de deelgraaf.

Opmerking
Het voert te ver om formeel te bewijzen dat dit algoritme in alle gevallen een minimaal opspannende boom van een samenhangende gewogen graaf berekent. Het algoritme van Kruskal wordt een 'gulzig' algoritme genoemd. Dit betekent dat je in elke stap lokaal de beste keuze maakt. Achteraf blijkt het algoritme van Kruskal ook globaal de beste keuze, hetgeen niet noodzakelijk betekent dat het de enige beste oplossing is.

Meer **wiskundig geformuleerd** luidt het algoritme van Kruskal als volgt.

Algoritme van Kruskal

Zij $G = (V, E)$ een samenhangende gewogen graaf van orde n. We construeren stap voor stap een opspannende boom H, startende van één boog met minimaal gewicht en voegen dan steeds bogen eraan toe die geen cykels doen ontstaan.

Beginstap
Neem een boog $e_1 \in E$ met minimaal gewicht en zij G_1 de deelgraaf van G bestaande uit de boog e_1 en zijn twee eindpunten. Pas de herhaalstap toe op G_1.

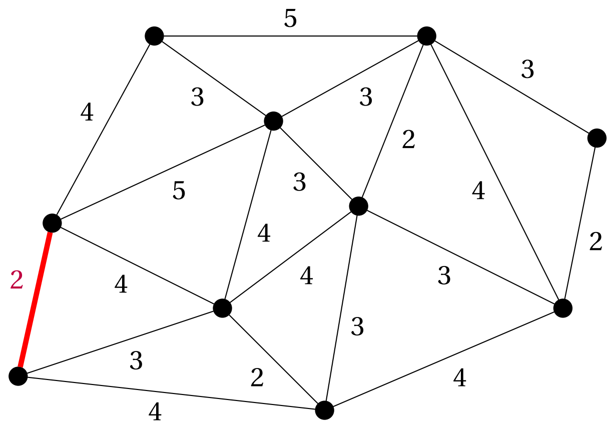
Herhaalstap
Stel dat we al een graaf G_i geconstrueerd hebben, met i tussen 1 en $n-2$, dus we kozen al bogen $e_1, \ldots, e_i$. Beschouw nu alle bogen uit E die:
- niet reeds gekozen zijn;
- als we ze toevoegen aan G_i dan ontstaan er geen cykels.

Neem als e_{i+1} één van deze bogen met minimaal gewicht.
We voegen de boog e_{i+1} en zijn eindpunten die nog niet behoorden tot G_i toe aan G_i, en noemen de bekomen graaf G_{i+1}.

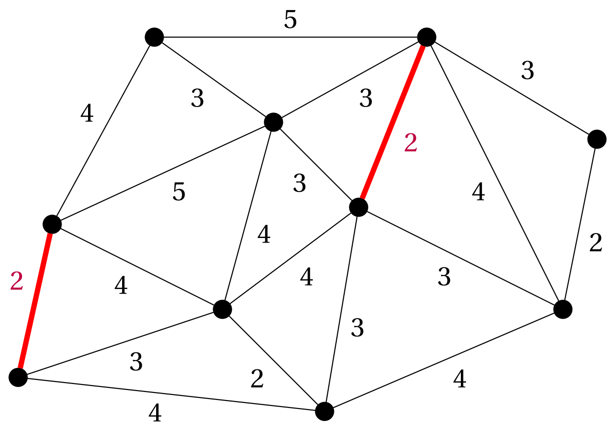
Als $i + 1 < n - 1$: pas de herhaalstap toe op G_{i+1};
als $i + 1 = n - 1$, stel $H = G_{i+1}$.

Op de volgende pagina zie je het algoritme toegepast op het inleidend voorbeeld.

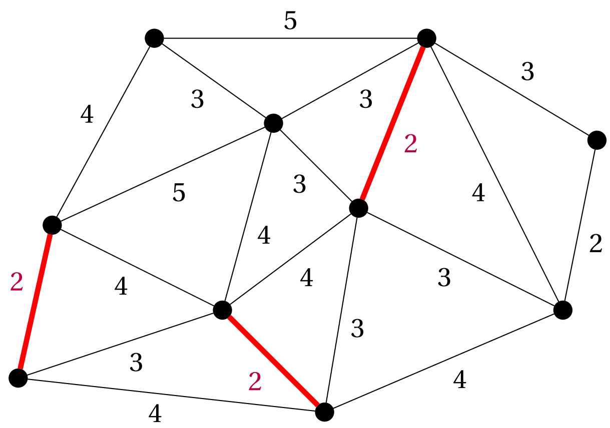
We starten met één van de bogen met laagste gewicht (2).



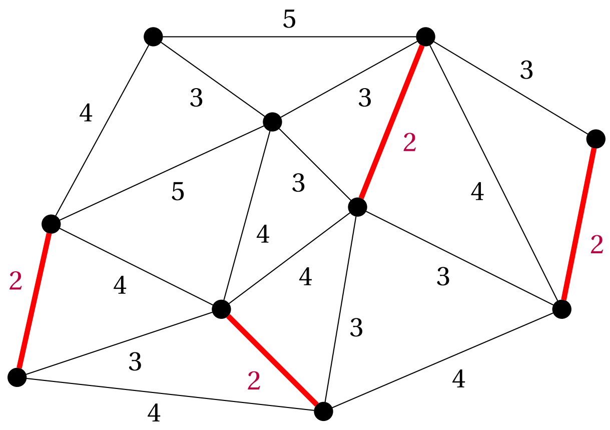
We voegen een boog toe met minimaal gewicht (2).



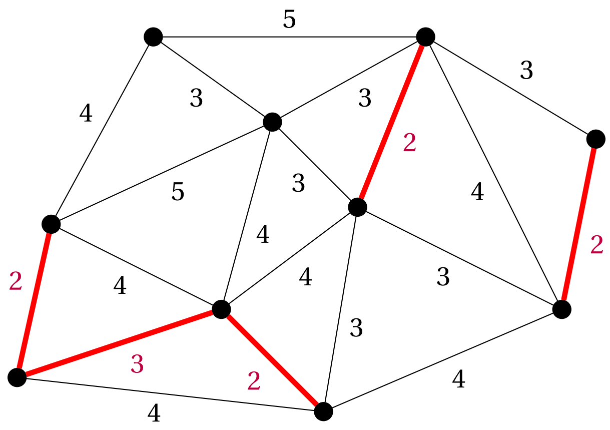
We voegen nog een boog toe met gewicht 2.



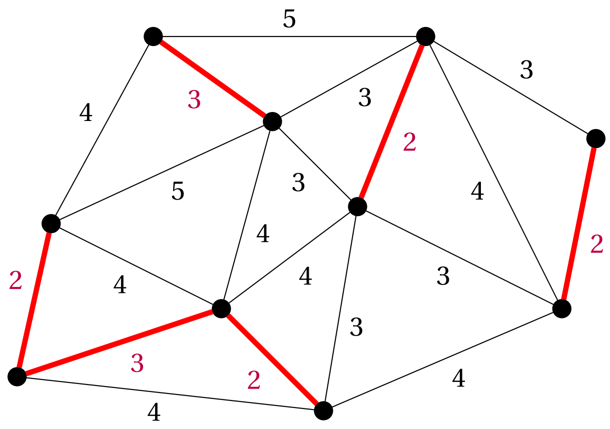
Er is nog een vierde boog met gewicht 2 die we kunnen toevoegen zonder dat er een cykel ontstaat.

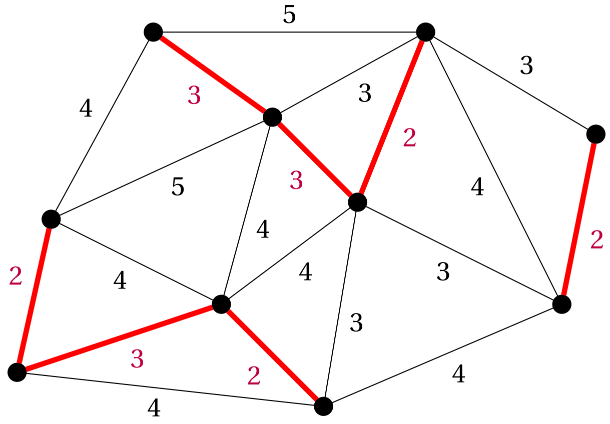


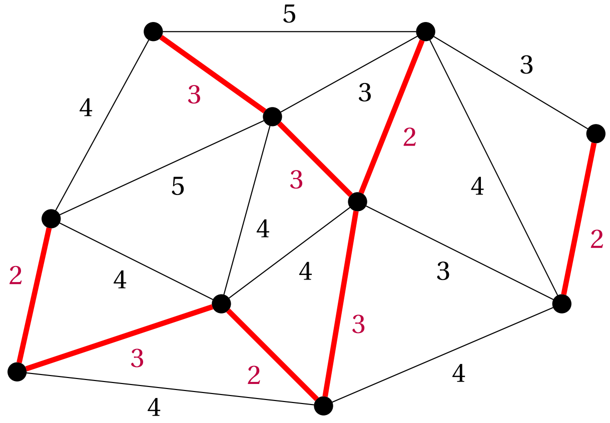
We kiezen één van de bogen met gewicht 3 en voegen deze toe.

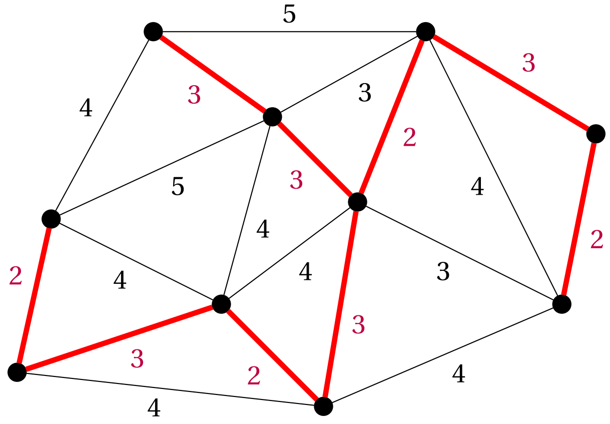


We voegen stap voor stap verder bogen met gewicht 3 toe, zonder dat er een cykel ontstaat.



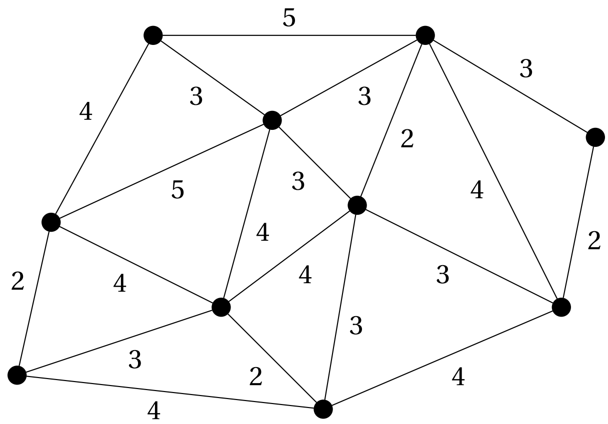






Als we nu nog een boog toevoegen, ontstaat een cykel. We hebben een minimaal opspannende boom gevonden. Het gewicht van deze deelgraaf is $2+2+2+2+3+3+3+3+3 = 23$. Met een bekabeling volgens dit plan legt een telecommunicatiebedrijf op de goedkoopste manier een netwerk aan in dit vakantieoord.

Opdracht
In sommige stappen van het algoritme zijn er meerdere mogelijkheden om een volgende boog te kiezen die kan worden opgenomen in de deelgraaf. Zoek een andere minimaal opspannende boom met gewicht 23 in deze graaf. Duid aan op onderstaande graaf.



4.4 Algoritme van Prim

Net zoals het algoritme van Kruskal is het algoritme van Prim een algoritme om een minimaal opspannende boom van een graaf te vinden.
Het algoritme werd in 1930 ontdekt door de wiskundige Jarník en in 1957 onafhankelijk herontdekt door de informaticus Prim.
Door het algoritme van Kruskal wordt telkens een boog met het laagste gewicht toegevoegd waarvoor geen cykel ontstaat, bij het algortime van Prim vertrekt men uit een willekeurige knoop en voegt telkens een boog met minimaal gewicht en knoop toe aan een op die manier groeiende minimaal opspannende boom.

Meer specifiek ga je als volgt te werk.

Algoritme van Prim

Ga uit van een samenhangende gewogen graaf $G = (V, E)$ met minstens 3 knopen.

1. **Begin**: kies een willekeurige knoop a.
   Noem N de verzameling van alle knopen die op elk moment tot de voorlopige oplossing behoren. Initieel bevat de verzameling N dus enkel het element a:
   $N_1 = \{a\}$.

2. **Herhaal** volgende stap: maak een lijst van alle bogen die een willekeurige knoop x, behorende tot de verzameling N, verbindt met een knoop die nog niet tot de verzameling N behoort.
   Kies uit die lijst de boog met het laagste gewicht. Zijn er meerdere opties – m.a.w. meerdere bogen met hetzelfde laagste gewicht – dan kies je één van die bogen. Deze boog maakt nu deel uit van de minimaal opspannende boom, en leidt naar een knoop y die nu toegevoegd wordt aan de verzameling N.

3. **Einde**: stop als alle knopen van G tot N behoren, m.a.w. als $N = V$.

Opmerkingen

1. Men kan formeel bewijzen dat ook het algoritme van Prim in alle gevallen een minimaal opspannende boom van een samenhangende gewogen graaf berekent. Dit algoritme is ook gulzig: er wordt telkens heel kortzichtig lokaal beslist welke boog nu toe te voegen (minste gewicht) zonder het globale doel in het oog te houden. Het bewijs van de correctheid van het algoritme van Prim toont aan dat de gulzigheid ook hier leidt tot optimaliteit.

2. Een wiskundige stelling over grafen is essentieel om de correctheid van een algoritme aan te tonen. Het bewijs gebruikt logica en zijn typische afleidingsregels, en dikwijls ook inductie. Een wiskundig inzicht leidt tot een efficiënt algoritme. En omgekeerd is een efficiënt algoritme gebaseerd op een wiskundig inzicht.

Meer wiskundig, en ook meer programmeerbaar, geformuleerd luidt het algoritme van Prim als volgt.

Algoritme van Prim

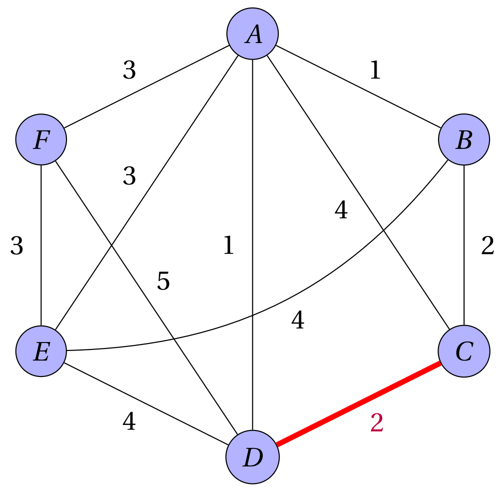
Zij $G = (V, E)$ een samenhangende gewogen graaf van orde n.
We construeren een deelgraaf $G' = (V, E')$ stap voor stap, zodanig dat de deelgraaf G' op elk moment een boom is (samenhangend en geen cykels).

Beginstap
Kies een knoop v_1 en neem een boog $e_1 \in E$ vertrekkend uit v_1 met minimaal gewicht.
Zij $G_1 = (V_1, E_1)$ de deelgraaf van G bestaande uit de boog e_1 en zijn twee eindknopen v_1 en v_1', dus $V_1 = \{v_1, v_1'\}$ en $E_1 = \{e_1\}$.
Pas de herhaalstap toe op G_1.

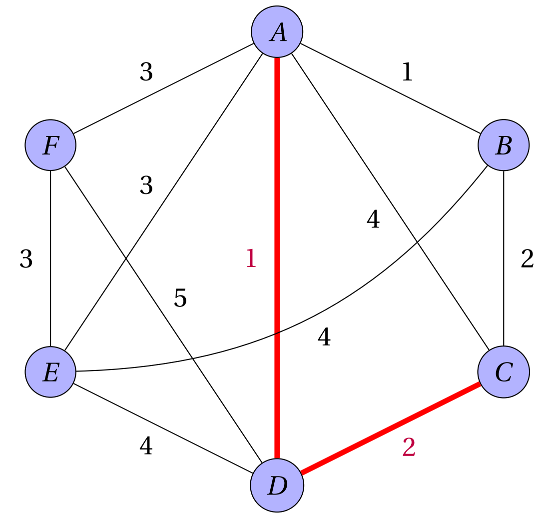
Herhaalstap
Stel dat we al een graaf $G_i = (V_i, E_i)$ geconstrueerd hebben, met i tussen 1 en $n-2$.
Beschouw alle bogen die een knoop van V_i verbinden met een knoop van V niet in V_i.
Neem hieruit een boog e_{i+1} met het laagste gewicht; zij v_{i+1} de eindknoop van e_{i+1} dat niet tot V_i behoort.
We voegen de boog e_{i+1} toe aan G_i.
Stel $E_{i+1} = E_i \cup \{e_{i+1}\}$ en $V_{i+1} = V_i \cup \{v_{i+1}\}$ en beschouw de graaf $G_{i+1} = (V_{i+1}, E_{i+1})$.

Als $i + 1 < n - 1$: pas de herhaalstap toe op G_{i+1};
als $i + 1 = n - 1$, stel $G' = G_{i+1}$.

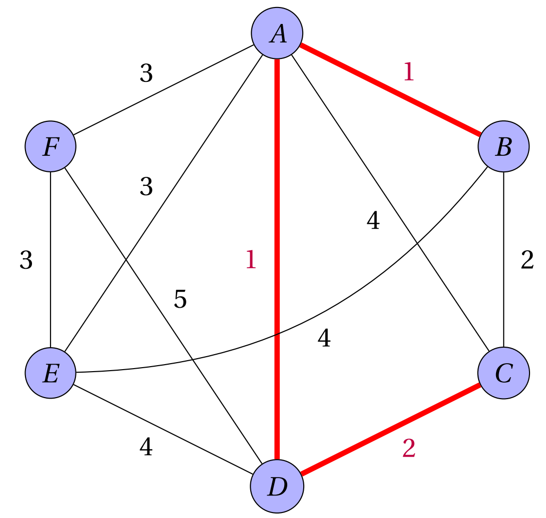
Hieronder zie je het algoritme toegepast op een voorbeeld.
We vertrekken we vanuit een willekeurige knoop van een gegeven graaf G: we kiezen C. Vervolgens bekijken we de bogen die vertrekken uit C. Het laagste gewicht is 2 en er zijn twee zulke bogen BC en CD. We kiezen boog CD.



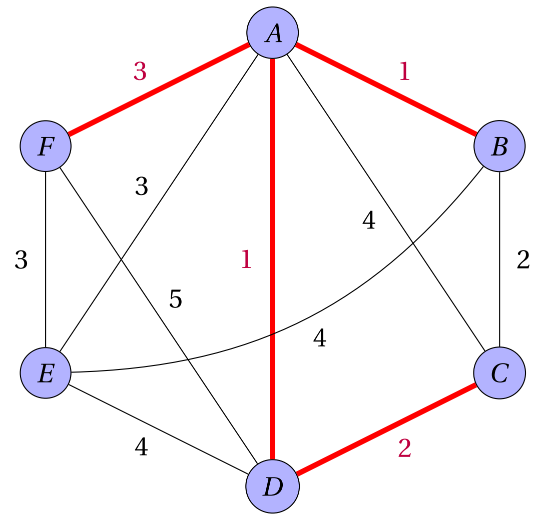
We bekijken nu alle bogen die vertrekken uit C en D en die nog niet gekozen zijn. Boog DA heeft van al deze bogen het kleinste gewicht (1), dus voegen we boog DA toe aan de deelgraaf.



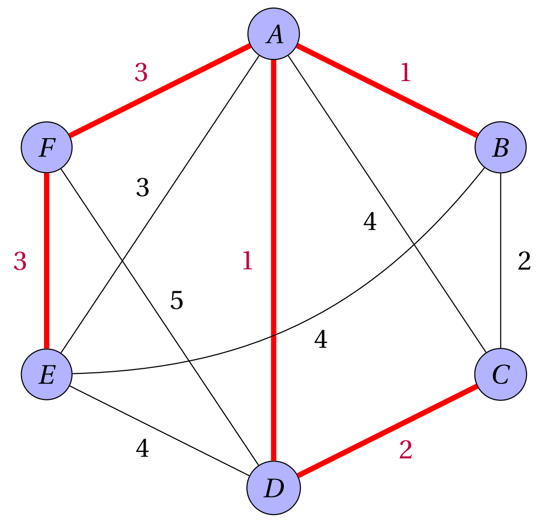
Boog AB is de boog met het kleinste gewicht (1) als we alle bogen beschouwen die een uiteinde hebben in één van de knopen C, D, A. We voegen dus boog AB toe.



Bogen AF en AE zijn de bogen met het kleinste gewicht (3) als we alle bogen beschouwen die een uiteinde hebben in één van de knopen C, D, A, B. We kiezen één van deze twee bogen, bv. AF, en voegen deze boog toe aan de boom.

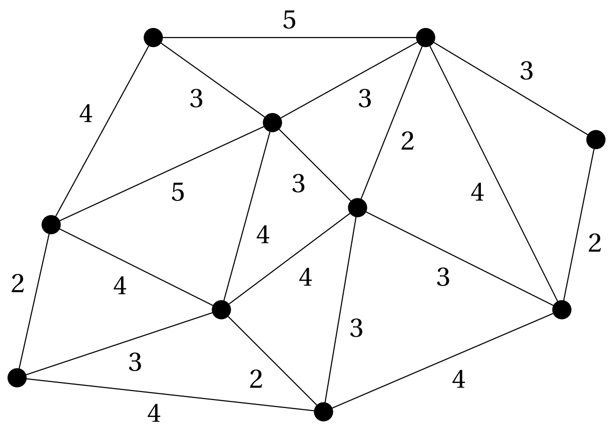


Bogen FE en AE zijn de bogen met het kleinste gewicht (3) als we alle bogen beschouwen die een uiteinde hebben in één van de knopen C, D, A, B, F. We kiezen één van deze twee bogen, bv. FE, en voegen deze boog toe aan de boom.



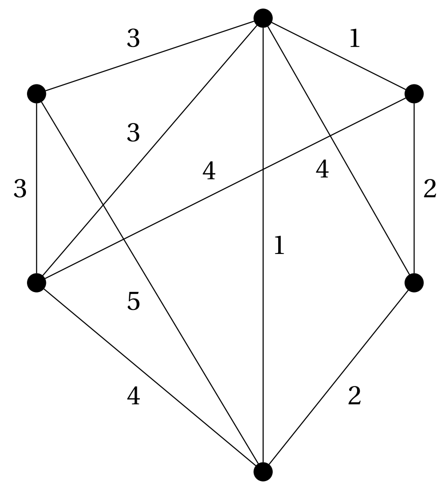
Alle knopen van G behoren tot de deelgraaf. We hebben een minimaal opspannende boom van G gevonden. Het gewicht van deze deelgraaf is 10.

Opdracht
Zoek met behulp van het algoritme van Prim het goedkoopste netwerk in het vakantieoord uit het inleidend voorbeeld. Duid het resultaat aan op onderstaande graaf.
Vergelijk het resultaat met de eerder gevonden goedkoopste netwerken met behulp van het algoritme van Kruskal.

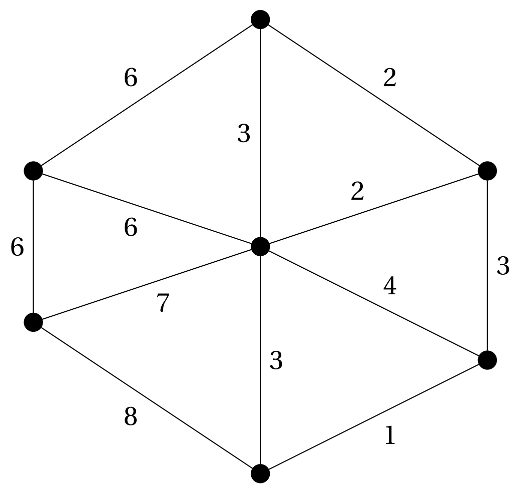


4.5 Oefeningen

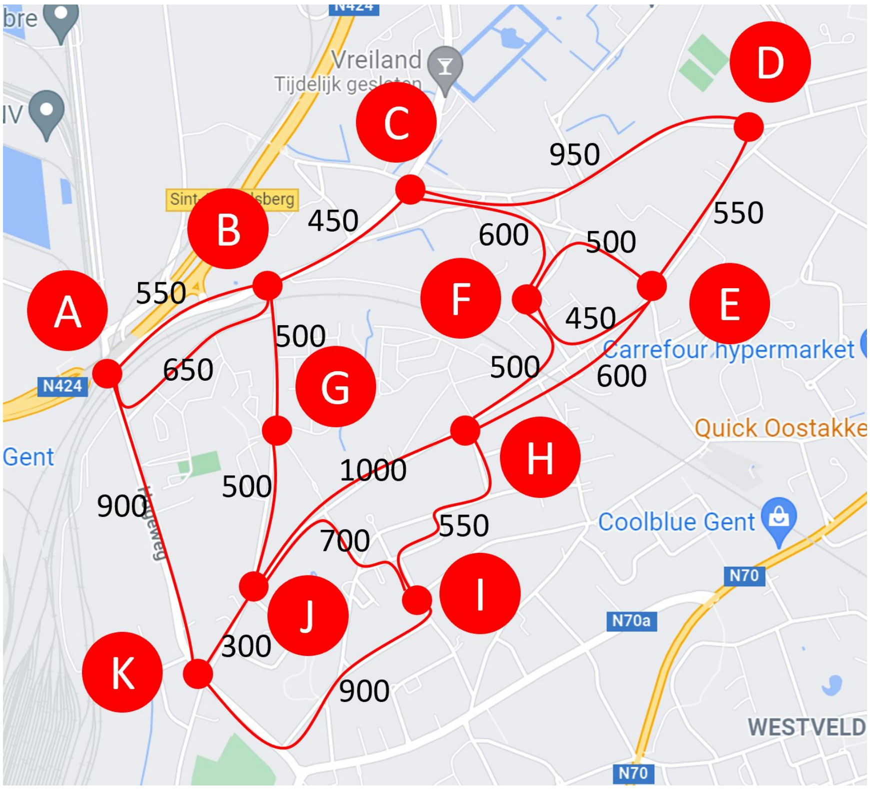
1. Teken alle mogelijke bomen met 5 knopen.
2. Zoek een minimaal opspannende boom in volgende graaf. Bepaal het gewicht van de gevonden minimaal opspannende boom.



3. Tussen 7 pc's, voorgesteld in onderstaande figuur, moet een netwerk worden aangelegd. De aanlegkosten van de verschillende verbindingen staan vermeld op de figuur (in duizenden euro's). Andere verbindingen dan de deze die zijn getekend zijn om technische reden niet mogelijk. Bereken de totale kosten van het goedkoopste netwerk.



4. Kato woont in Oostakker, een deelgemeente van Gent. Jammer genoeg zijn er in Oostakker nauwelijks fietspaden wat het heel onveilig maakt voor Kato om af te spreken met haar 10 vriendinnen, die ook allemaal in Oostakker wonen.
   Ze droomt ervan om een netwerk van fietspaden te hebben die haar en haar 10 vriendinnen onderling verbindt. Samen werken ze aan een actieplan waarbij ze op zoek gaan naar een zo goedkoop mogelijk netwerk van fietspaden. Met dit goedkoopste plan hopen ze gehoor te krijgen bij het stadsbestuur van Gent. Ze nemen Google Maps erbij en duiden het huis van elke vriendin aan op de kaart. Vervolgens meten ze, met de routeplanner van Google Maps, de afstand tussen elk huis. Voor sommige verbindingen zijn er meerdere mogelijkheden, ze noteren het allemaal op de kaart.



   Teken de bijhorende graaf en bepaal het goedkoopste netwerk van fietspaden voor Kato en haar vriendinnen. Duid dit netwerk aan op de graaf en op de kaart.

5. In één van de zes wijken (A, B, C, D, E en F) van een stad moet een dokterspost komen. In onderstaande tabel staan tussen haakjes steeds twee getallen. Het eerste getal is de afstand (in km) tussen de centra van twee wijken. Het tweede getal geeft de gemiddelde tijd (in minuten) aan die nodig is om van het centrum van de ene wijk naar het centrum van de andere wijk te komen. Een liggend streepje geeft aan dat er geen rechtstreekse weg is tussen de twee wijken.

	B	C	D	E	F
A	(7,1 ; 3,5)	–	–	(7,3 ; 3,3)	(6,8 ; 2,9)
B		(6,6 ; 3,3)	(6,5 ; 2,8)	(5,8 ; 2,7)	–
C			(6,1 ; 2,9)	(6,2 ; 3,6)	–
D				(7,0 ; 3,5)	–
E					(6,5 ; 3,1)

Er zijn twee gegevens waar naar gekeken kan worden, namelijk de afstand en de tijd. In de wijk die het meest centraal gelegen is, komt de dokterspost.

(a) Bepaal door een minimaal opspannende boom te tekenen, de meest centrale wijk(en) wanneer men let op afstand.

(b) Bepaal door een minimaal opspannende boom te tekenen, de meest centrale wijk(en) wanneer men let op tijd.

Module 5

Dijkstra

5.1 Kortste pad

Hoe bepaalt een GPS je route? Een kortste pad van het ene naar het andere punt zoeken is een probleem dat regelmatig moet worden opgelost, en liefst zo efficiënt mogelijk.
Een navigatiesysteem gebruikt een goed kortste-pad-algoritme[1]: je geeft je bestemming in en je krijgt de kortste weg van je huidige positie naar je bestemming. Soms betekent 'kortste' dat dit de weg is met een minimaal aantal kilometers. Afhankelijk van de instellingen of voorkeuren, kan 'kortste' betekenen dat de voorgestelde weg het minste tijd vraagt, of het minste kost aan tolgeld, of de veiligste weg is, of de weg is langs toeristische bezienswaardigheden...
Voor toeristen is een GPS plezant, voor professionele verplaatsingen en voor vervoersmaatschappijen is een GPS van groot economisch belang. Ook moeten routes snel herberekend kunnen worden als de omstandigheden veranderen: files, onderbroken wegen...
Allemaal goede redenen om efficiënte kortste-pad-algoritmen te bestuderen.

5.2 De reisplanner

We starten met een voorbeeld. Je bent op vakantie aan zee in Middelkerke. Je beste vriendin trouwt met de partner van haar dromen in de Ardennen, in Barvaux. Je bent uitgenodigd voor het huwelijksfeest en bekijkt hoe je met het openbaar vervoer van Middelkerke naar Barvaux kan reizen. Hoeveel keer en waar moet je overstappen? Wat is de snelste route? Wat is de kostprijs van de reis? Op al deze vragen vind je antwoord op de website van de NMBS (https://www.belgiantrain.be). Je krijgt meerdere opties. Je reisschema kan er als volgt uitzien:

Middelkerke Kerk [De Lijn]	V	12:22
Oostende Station	A	12:45
Oostende Station	V	13:10
Gent-Sint-Pieters	A	13:49
Gent-Sint-Pieters	V	13:55
Brussel-Zuid	A	14.24
Brussel-Zuid	V	14:33
Marloie	A	16:16
Marloie	V	16:35
Barvaux	A	16:52

De reistijd bedraagt in dit geval 4u30min en je moet 4 keer overstappen. Een andere mogelijkheid is:

[1]Een algoritme is een vindregel, m.a.w. een vaste stap voor stap werkwijze die leidt tot de oplossing van een probleem, bv. methode om een staartdeling uit te voeren, discriminantmethode voor het oplossen van tweedegraadsvergelijkingen...

Middelkerke Kerk [De Lijn]	V	12:22
Oostende Station	A	12:45
Oostende Station	V	13:43
Luik-Guillemins	A	15:59
Luik-Guillemins	V	16:16
Barvaux	A	17:05

Met deze keuze ben je iets langer onderweg (4u43min), maar je moet slechts 2 maal overstappen.
De kosten van de reis zijn dezelfde (21,80 EUR).

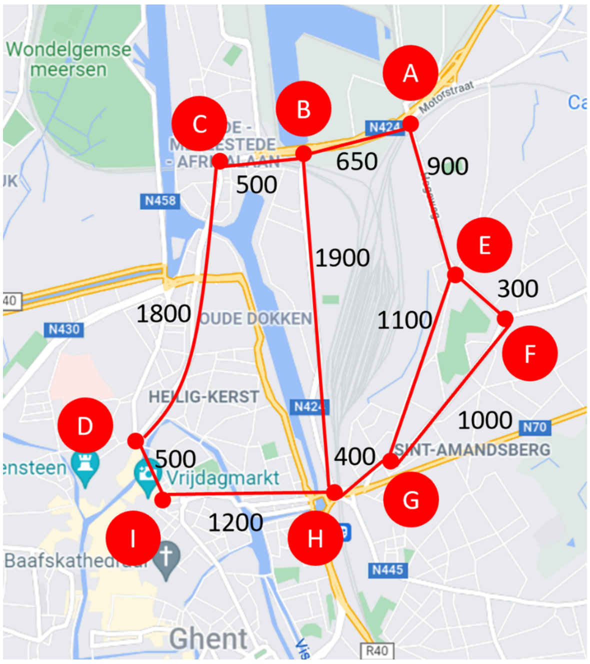
De Belgische Spoorwegen hebben een reisplanner op internet. Je hoeft enkel begin- en eindpunt van je reis in te geven, plus vertrek- of aankomsttijd, en vervolgens krijg je mogelijke routes. Maar hoe werkt zo'n reisplanner? Hoe worden deze routes bepaald?
Eén zo'n algoritme dat de kortste, snelste of goedkoopste route tussen twee willekeurige plaatsen bepaald, is het algoritme van Dijkstra.

5.3 Algoritme van Dijkstra - een stukje geschiedenis

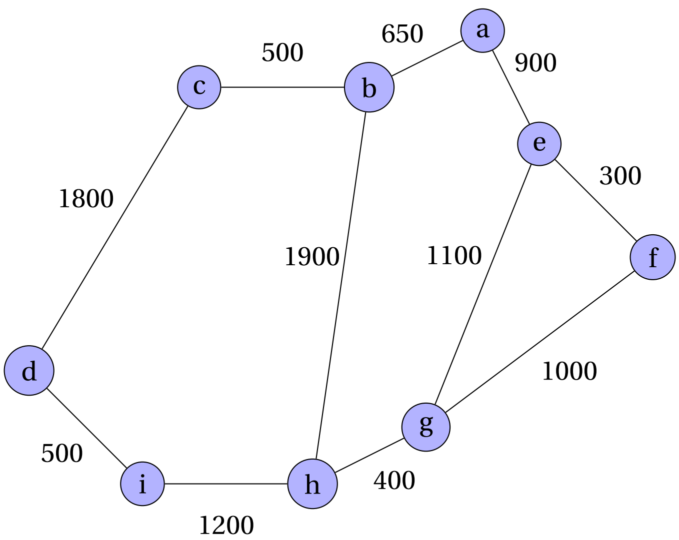
In 1956 bedacht Edsger Dijkstra (1930-2002) zijn kortste pad-algoritme. Het werd gepubliceerd 3 jaar later. Dijkstra was één van de eerste programmeurs in Nederland. Toen hij in 1957 trouwde, bleek hoezeer Dijkstra zich in de voorhoede bewoog: het beroep dat hij opgaf – computerprogrammeur – werd door de burgerlijke stand niet erkend. Omdat hij natuurkunde had gestudeerd, gaf hij toen maar 'theoretisch natuurkundige' op.
Zijn beroemdste resultaat is het kortstepad-algoritme, dat de kortste verbinding vindt tussen twee knopen in een samenhangende gewogen graaf. Het slimme van Dijkstra's algoritme is dat het niet alle mogelijke routes met elkaar vergelijkt, maar dat het stap voor stap de kortst mogelijke afstanden tot elke knoop opbouwt. In de eerste stap kijk je naar alle knopen die vanaf het beginpunt te bereiken zijn en markeer je al die knopen met de afstand tot de beginknoop. Daarna kijk je steeds vanaf de knoop dat op dat moment de kortste afstand heeft tot de beginknoop naar alle knopen die je vanaf daar kunt bereiken. Als je een naburige knoop via een nieuwe verbinding op een snellere manier kan bereiken, schrijf je de nieuwe, kortere afstand tot de beginknoop bij die knoop. Zo ga je steeds een stukje verder tot je alle knopen hebt gehad en je de kortste route tot de eindknoop hebt gevonden. We bekijken dit algoritme hierna in detail. Je zal merken dat het zo eenvoudig is dat je je afvraagt waarom je het niet zelf hebt bedacht. Dijkstra vond het zelf ook een beetje gek dat zijn naam en faam voor een groot deel gebaseerd waren op een algoritme dat hij bedacht als demonstratie voor een computer. Hij bedacht het kortstepadalgoritme zonder pen of papier op een zonnig terras terwijl hij met zijn verloofde een kopje koffie dronk.

5.4 Algoritme van Dijkstra - voorbeeld

Kato woont in Gent aan het kruispunt van de Hogeweg met de Motorstraat (a) en wil gaan shoppen in de omgeving van de Vrijdagmarkt (i) in het centrum van Gent. Ze mag van haar mama met de fiets gaan maar moet op voorhand opzoeken hoe ze zal rijden. Ze wil immers de kortste weg nemen.



Kato zoekt dus het kortste pad van ‘a’ naar ‘i’ in onderstaande graaf.



We starten in knoop a. We kennen aan elke knoop x twee labels toe:

- het eerste label $l(x)$ geeft aan wat tijdens de stappen in het algoritme de tot dan toe berekende benadering van de kortste afstand is tussen startknoop a en knoop x
- het tweede label $v(x)$ geeft aan welke knoop daarbij de directe voorganger is van knoop x op de dan toe berekende benadering van het kortste pad van a naar x

De labels worden gedurende het algoritme steeds aangepast zodat ze op het einde van het algoritme de gevraagde kortste afstand geven en de informatie om het kortste pad terug te vinden.

We noteren het gewicht van boog xy als $w(xy)$.

Algoritme van Dijkstra

Ga uit van een samenhangende gewogen graaf. Bepaal kortste pad van a naar b als volgt.

1. **Begin**: stel $l(a) = 0$; $v(a) = a$ en $l(p) = +\infty$ voor alle knopen $p \neq a$.
   Maak een lijst met afgehandelde knopen en zet hier knoop a in.
2. **Herhaal**:
   - stel u is de laatste knoop die aan de lijst met afgehandelde knopen is toegevoegd
   - bekijk de niet afgehandelde naburige knopen p van u
     - als $l(p) > l(u) + w(up)$, dan $l(p) = l(u) + w(up)$ en $v(p) = u$
     - als $l(p) \leqslant l(u) + w(up)$, behoud $l(p)$ en $v(p)$
   - Kies van alle niet afgehandelde knopen de knoop met de laagste waarde voor $l(p)$. Voeg deze knoop toe aan de lijst met afgehandelde knopen.
3. **Einde**: als knoop b in de lijst met afgehandelde knopen komt, dan stoppen. Je hebt het kortste pad van a naar b gevonden.

We passen deze werkwijze toe op het voorbeeld en houden de waarden voor $l(p)$ en $v(p)$ elke stap bij in een tabel.

Begin

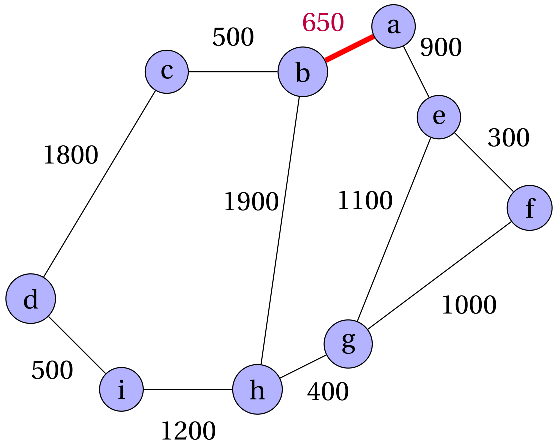
We noteren volgende labels: $l(a) = 0$, $v(a) = a$ en $l(p) = +\infty$ voor alle knopen $p \neq a$. We plaatsen a in de lijst met afgehandelde knopen. Het tweede label $v(p)$ voor $p \neq a$ kunnen we nog niet invullen.

	a	b	c	d	e	f	g	h	i	afgehandeld
Stap	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	
0	$0;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a

Herhaalstap 1

b en e zijn naburige knopen van a, afstanden tot a zijn kleiner dan $+\infty$ dus $l(b) = 650$ en $l(e) = 900$. Voorganger van b en e is a, dus $v(b) = a$ en $v(e) = a$. Laagste waarde is $l(b)$, dus we plaatsen b bij de afgehandelde knopen.

	a	b	c	d	e	f	g	h	i	afgehandeld
Stap	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	
0	$0;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a
1		$650;a$	$+\infty$;-	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a,b



Herhaalstap 2

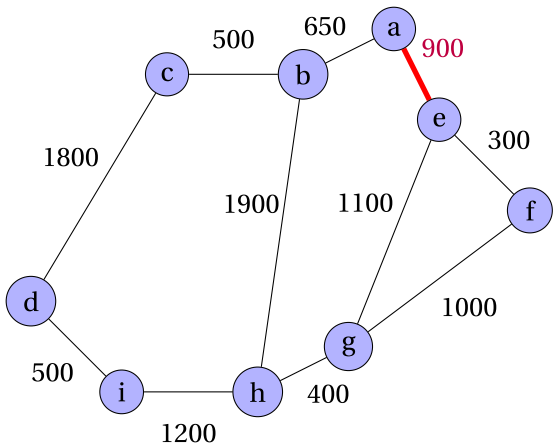
c en h zijn naburige knopen van b.

Afstand van a naar c via b is $650 + 500 = 1150$, dit is kleiner dan $+\infty$, dus wordt $l(c) = 1150$ en $v(c) = b$.

Afstand van a naar h via b is $650 + 1900 = 2550$, dit is kleiner dan $+\infty$, dus wordt $l(h) = 2550$ en $v(h) = b$.

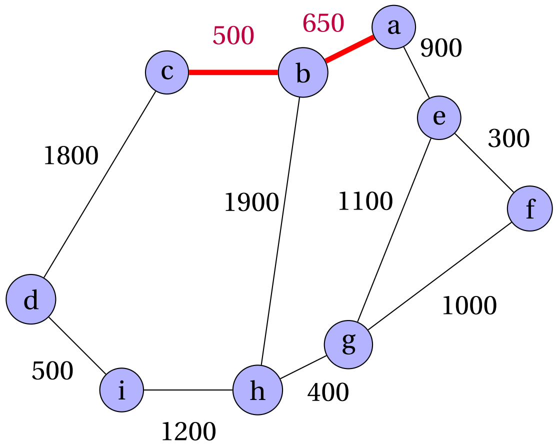
Van alle niet afgehandelde knopen heeft $l(e)$ de kleinste waarde, dus we voegen e toe aan de lijst afgehandelde knopen.

	a	b	c	d	e	f	g	h	i	afgehandeld
Stap	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	
0	$0;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a
1		$650;a$	$+\infty$;-	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a,b
2			$1150;b$	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$2550;b$	$+\infty$;-	a,b,e



Herhaalstap 3
f en g zijn naburige knopen van e.
Afstand van a naar f via e is $900+300=1200$, dit is kleiner dan $+\infty$, dus wordt $l(f)=1200$ en $v(f)=e$.
Afstand van a naar g via e is $900+1100=2000$, dit is kleiner dan $+\infty$, dus wordt $l(g)=2000$ en $v(g)=e$.
Van alle niet afgehandelde knopen heeft $l(c)$ de kleinste waarde, dus we voegen c toe aan de lijst afgehandelde knopen.

	a	b	c	d	e	f	g	h	i	afgehandeld
Stap	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	
0	$0;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a
1		$650;a$	$+\infty$;-	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a,b
2			$1150;b$	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$2550;b$	$+\infty$;-	a,b,e
3			$1150;b$	$+\infty$;-		$1200;e$	$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c



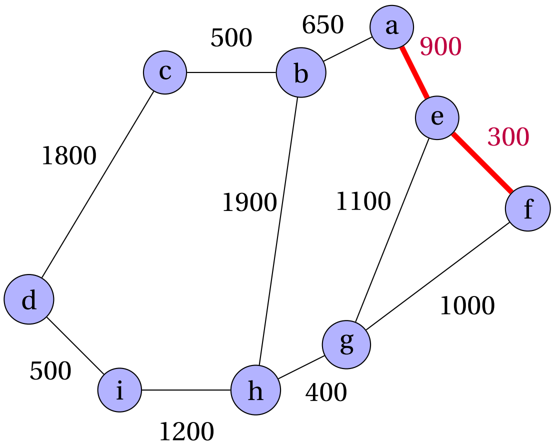
Herhaalstap 4

d is naburig aan c.

Afstand van a naar d via b en c is $1150 + 1800 = 2950$, dit is kleiner dan $+\infty$, dus wordt $l(d) = 2950$ en $v(d) = c$.

Van alle niet afgehandelde knopen heeft $l(f)$ de kleinste waarde, dus we voegen f toe aan de lijst afgehandelde knopen.

	a	b	c	d	e	f	g	h	i	afgehandeld
Stap	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	
0	$0;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a
1		$650;a$	$+\infty$;-	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a,b
2			$1150;b$	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$2550;b$	$+\infty$;-	a,b,e
3			$1150;b$	$+\infty$;-		$1200;e$	$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c
4				$2950;c$		$1200;e$	$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c,f



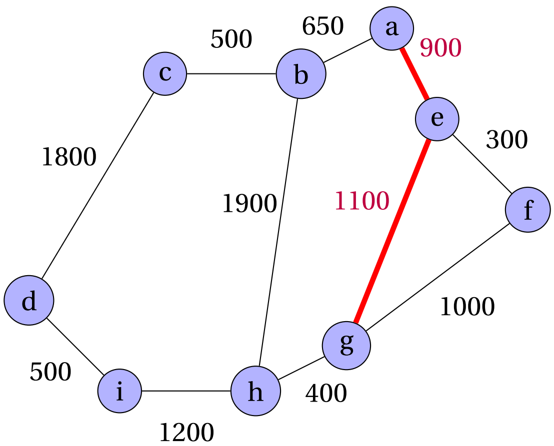
Herhaalstap 5

g is naburig aan f.

Afstand van a naar g via e en f is $1200+1000 = 2200$, dit is groter dan $l(g) = 2000$, dus behouden we $l(g) = 2000$ en $v(g) = e$.

Van alle niet afgehandelde knopen heeft $l(g)$ de kleinste waarde, dus we voegen g toe aan de lijst afgehandelde knopen.

	a	b	c	d	e	f	g	h	i	afgehandeld
Stap	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	
0	$0;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a
1		$650;a$	$+\infty$;-	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a,b
2			$1150;b$	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$2550;b$	$+\infty$;-	a,b,e
3			$1150;b$	$+\infty$;-		$1200;e$	$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c
4				$2950;c$		$1200;e$	$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c,f
5				$2950;c$			$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c,f,g



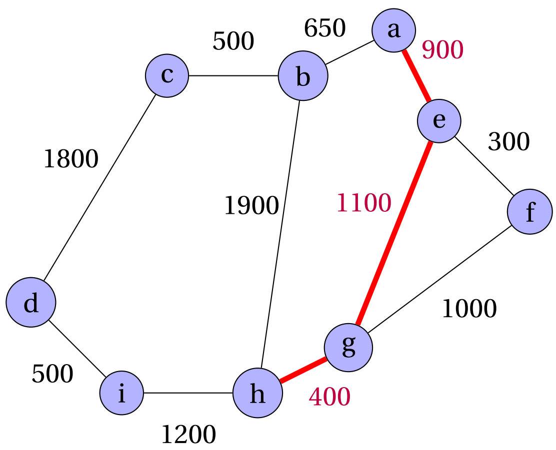
Herhaalstap 6

h is naburig aan g.

Afstand van a naar h via e, f en g is $2000 + 400 = 2400$, dit is kleiner dan $l(h) = 2550$, dus wordt $l(h) = 2400$ en $v(h) = g$.

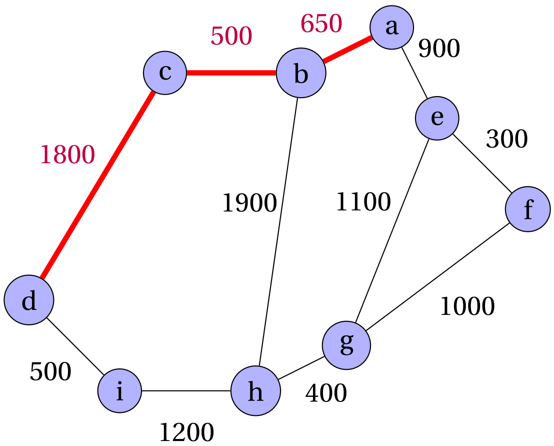
Van alle niet afgehandelde knopen heeft $l(h)$ de kleinste waarde, dus we voegen h toe aan de lijst afgehandelde knopen.

	a	b	c	d	e	f	g	h	i	afgehandeld
Stap	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	
0	$0;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a
1		$650;a$	$+\infty$;-	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a,b
2			$1150;b$	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$2550;b$	$+\infty$;-	a,b,e
3			$1150;b$	$+\infty$;-		$1200;e$	$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c
4				$2950;c$		$1200;e$	$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c,f
5				$2950;c$			$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c,f,g
6				$2950;c$				$2400;g$	$+\infty$;-	a,b,e,c,f,g,h



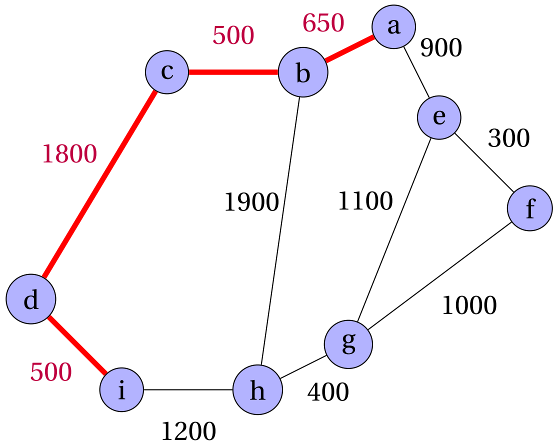
Herhaalstap 7
i is naburig aan h.
Afstand van a naar i via e, g en h is $2400+1200=3600$, dit is kleiner dan $+\infty$, dus wordt $l(i)=3600$ en $v(i)=h$.
Van alle niet afgehandelde knopen heeft $l(d)$ de kleinste waarde, dus we voegen d toe aan de lijst afgehandelde knopen.

	a	b	c	d	e	f	g	h	i	afge-han-deld
Stap	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	
0	$0;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a
1		$650;a$	$+\infty$;-	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a,b
2			$1150;b$	$+\infty$;-	$900;a$	$+\infty$;-	$+\infty$;-	$2550;b$	$+\infty$;-	a,b,e
3			$1150;b$	$+\infty$;-		$1200;e$	$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c
4				$2950;c$		$1200;e$	$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c,f
5				$2950;c$			$2000;e$	$2550;b$	$+\infty$;-	a,b,e,c,f,g
6				$2950;c$				$2400;g$	$+\infty$;-	a,b,e,c,f,g,h
7				$2950;c$					$3600;h$	a,b,e,c,f,g,h,d



Herhaalstap 8 = Einde
i is naburig aan d.
Afstand van a naar i via b, c en d is $2950 + 500 = 3450$, dit is kleiner dan $l(i) = 3600$, dus wordt $l(i) = 3450$ en $v(i) = d$.
We kunnen nu knoop i als laatste knoop toevoegen aan de lijst met niet afgehandelde knopen. Hiermee hebben we de eindknoop bereikt. In de tabel kunnen we aflezen dat de eindknoop i wordt bereikt langs het kortste pad a, b, c, d, i. De kortste afstand tussen a en i is 3450.
Kato moet dus 3,45 km fietsen langs b, c en d om op de Vrijdagmarkt te geraken.

	a	b	c	d	e	f	g	h	i	afgehandeld
Stap	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	$(l;v)$	
0	$0; a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a
1		$650; a$	$+\infty$;-	$+\infty$;-	$900; a$	$+\infty$;-	$+\infty$;-	$+\infty$;-	$+\infty$;-	a, b
2			$1150; b$	$+\infty$;-	$900; a$	$+\infty$;-	$+\infty$;-	$2550; b$	$+\infty$;-	a, b, e
3			$1150; b$	$+\infty$;-		$1200; e$	$2000; e$	$2550; b$	$+\infty$;-	a, b, e, c
4				$2950; c$		$1200; e$	$2000; e$	$2550; b$	$+\infty$;-	a, b, e, c, f
5				$2950; c$			$2000; e$	$2550; b$	$+\infty$;-	a, b, e, c, f, g
6				$2950; c$				$2400; g$	$+\infty$;-	a, b, e, c, f, g, h
7				$2950; c$					$3600; h$	a, b, e, c, f, g, h, d
8									$3450; d$	$a, b, e, c, f, g, h, d, i$



Opmerkingen

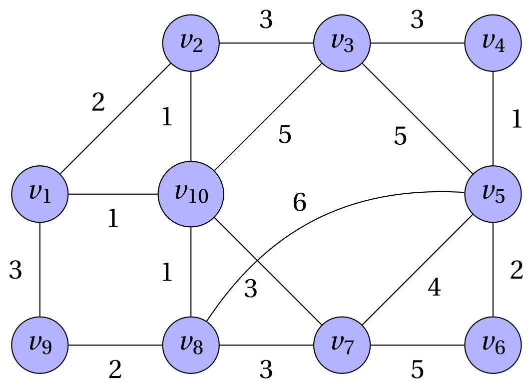
1. In dit geval zijn met het bereiken van de eindknoop ook alle knopen afgehandeld. Dit zal niet steeds het geval zijn, maar als je de herhaalstap blijft toepassen totdat alle knopen zijn afgehandeld, dan kan je voor elke knoop het kortste pad vinden vanuit de startknoop. Zowel het pad (via het tweede label voorganger $v(x)$) als de kortste afstand (via eerste label $l(x)$) kunnen worden afgelezen uit de tabel.

2. In het behandelde voorbeeld is het een vrij groot karwei om via het algoritme van Dijkstra het kortste pad te bepalen. Wellicht had je op het zicht sneller dit kortste pad kunnen vinden. Als er echter meer knopen en meer bogen zijn, dan wordt het een stuk lastiger om door trial en error de oplossing te vinden. Het algoritme van Dijkstra biedt dan uitkomst. Bovendien kunnen computerprogramma's dit algoritme in een minimum van tijd uitvoeren, zelfs voor zeer grote netwerken. Het snel kunnen vinden van kortste paden in netwerken was van essentieel belang voor ontwikkelingen zoals internet en GPS-systemen.

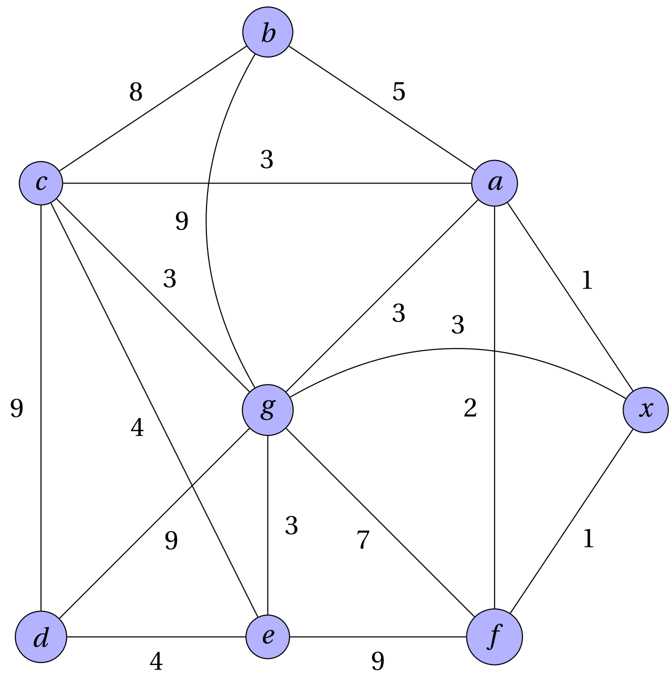
3. Het algoritme van Dijkstra is net zoals de algoritmes van Kruskal en van Prim een gulzig algoritme.

5.5 Oefeningen

1. Bepaal m.b.v. het algoritme van Dijkstra het kortste pad tussen de knopen v_1 en v_5 in onderstaande graaf.



2. Pas het algoritme van Dijkstra toe op onderstaande graaf, om uitgaande van knoop x het kortste pad te bepalen tot elke andere knoop. Geef ook het gewicht van die kortste paden.



3. Een bedrijf met hoofdzetel in Brussel (B) heeft bovendien 5 vestigingen in Gent (G), Den Haag (D), Amsterdam (A), Luxemburg (L) en Rijsel (R). Het bedrijf beschikt over abonnementen voor enkele verbindingen per trein tussen de verschillende vestigingen. De treinreiskosten tussen deze steden staan in onderstaande tabel.

	G	A	D	L	R
B	9	82	35	22	30
G			43	22	26
A			14	90	
L					96

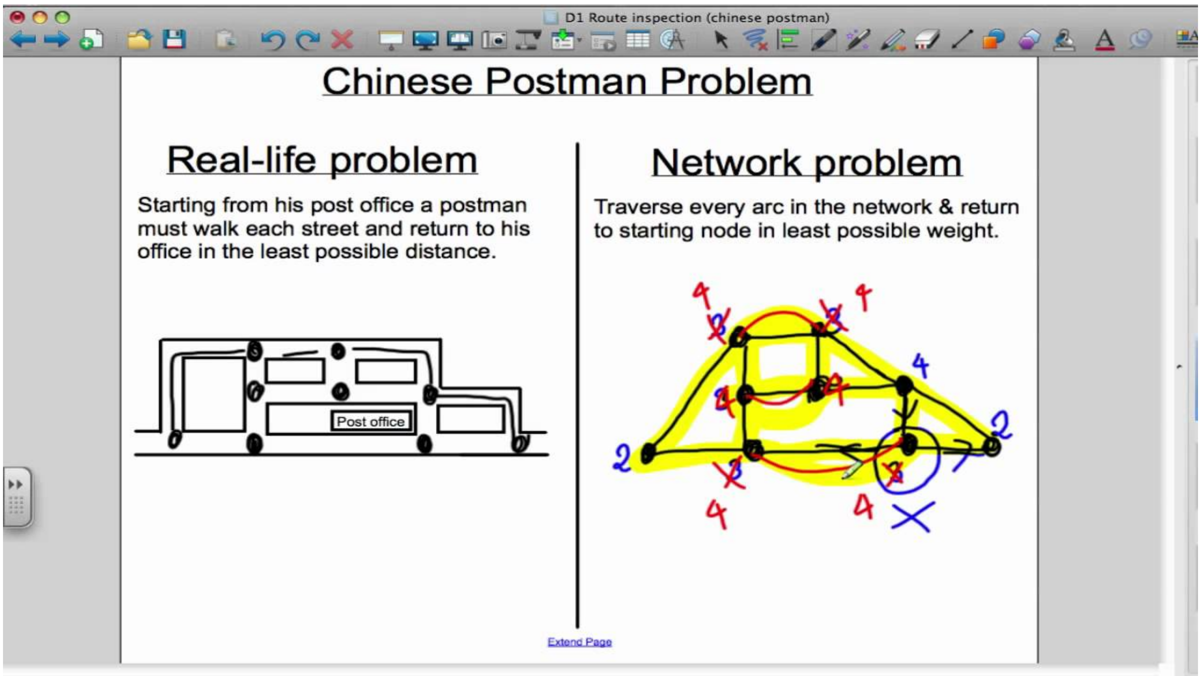
   (a) Stel deze tabel voor d.m.v. een gewogen graaf.

   (b) Bepaal m.b.v. het algoritme van Dijkstra de goedkoopste route tussen Rijsel en Amsterdam.

   (c) Bepaal ook de goedkoopste route tussen Luxemburg en Amsterdam.

Module 6

Postbodes

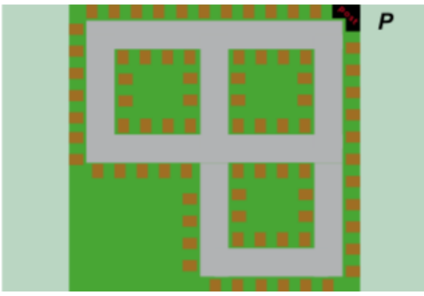


6.1 Het Chinees postbodeprobleem

Bij het handelsreizigers probleem wordt een optimale route gezocht waarbij alle plaatsen (knopen) moeten worden aangedaan.
Een postbode daarentegen zoekt de kortste route die ten minste eenmaal door alle straten (bogen), waar hij of zij post moet bezorgen, gaat. Deze route start en begint bij een postkantoor.
Dit probleem staat bekend als het Chinese Postbode Probleem. Chinees, omdat het probleem werd opgelost door de Chinese wiskundige Mei-Ko Kwan in 1962.

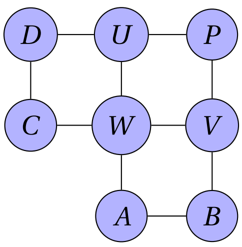
6.2 Een eerste eenvoudig voorbeeld

In een minidorpje waarvan je de plattegrond hieronder ziet, moet de postbode elke straat ten minste één keer doorlopen. Hij begint en eindigt in het postkantoor P en wil zó lopen, dat de totale afgelegde afstand zo klein mogelijk is.

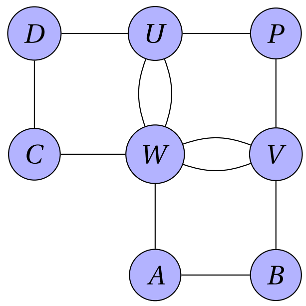


De postbode heeft geluk als op alle kruispunten een even aantal straten samenkomen. Immers, vertaald naar de bijbehorende graaf betekent dat dit elke knoop een even graad heeft. In dat geval weten we ondertussen dat er een Euleriaans circuit en is het antwoord op het vraagstuk gemakkelijk (zie module 1). De postbode doorloopt elke straat precies eenmaal, daardoor is dit zeker ook de kortste route.
Is er een dichte bebouwing en staan er huizen aan beide kanten van de straat, dan kan de postbode het Euleriaans circuit tweemaal afleggen: eerst de huizen met oneven huisnummers en daarna de huizen met even huisnummers (of omgekeerd).

Als we de eenvoudige plattegrond ontdoen van franje, dan krijgen we horend bij deze eenvoudige plattegrond volgende graaf.



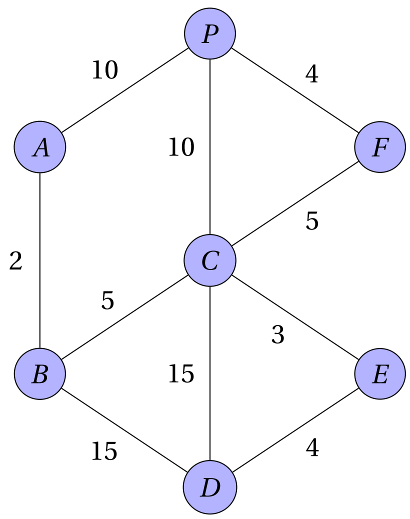
Er bestaat geen Eulercircuit in deze graaf die start en eindigt bij het postkantoor P, omdat knopen U en V een oneven graad hebben.
Deze knopen moeten een even graad krijgen, dus we verbinden ze met elkaar. Dat lukt niet rechtstreeks, omdat er geen straat loopt tussen de kruispunten U en V. We moeten dus via een ander kruispunt, dus andere knoop, U en V verbinden. Dit betekent dat de postbode twee straten tweeemaal moet doorlopen. Bv. als volgt.



Opdracht: duid door nummering van de bogen een Eulercircuit aan die weergeeft hoe de postbode zijn route eruit kan zien.

6.3 Een meer realistisch voorbeeld

Het simpele stratenplan uit vorig voorbeeld is niet erg realistisch: behalve dat het om een minidorpje gaat, is elke straat even lang. We krijgen een uitdagender probleem als niet elke straat een gelijke lengte heeft, of als niet elke straat evenveel tijd vergt om doorlopen te worden. Dan krijgen we gewogen graaf als voorstelling van een dergelijk stratenplan. Bv.



De getallen (gewichten) stellen de tijd in minuten voor om de straat door te lopen. De plaats van het postkantoor wordt weergegeven door knoop P.

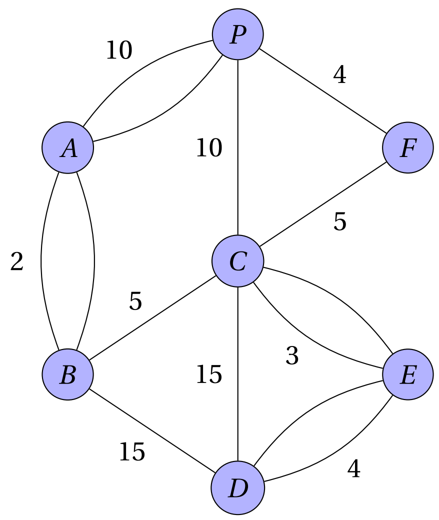
Deze graaf heeft vier knopen met oneven graad: P, B, C, D. We kunnen deze vier knopen op drie manieren paarsgewijs koppelen om extra verbindingen aan te leggen, zodat alle knopen een even graad krijgen. We willen dit op een zodanige manier doen dat we een kortste pad bekomen. Daarom bepalen we eerst het kortste pad tussen elk tweetal knopen met oneven graad.

- van P naar B: de wandeling (P, A, B) heeft lengte 12
- van P naar C: de wandeling (P, F, C) heeft lengte 9
- van P naar D: de wandeling (P, F, C, E, D) heeft lengte 16
- van B naar C: de wandeling (B, C) heeft lengte 5
- van B naar D: de wandeling (B, D) heeft lengte 15
- van C naar D: de wandeling (C, E, D) heeft lengte 7

Mogelijke paarsgewijze verbindingen leveren volgende extra afstanden dus toe:

- PB en CD: $12 + 7 = 19$
- PC en BD: $9 + 15 = 24$
- PD en BC: $16 + 5 = 21$

De eerste optie levert de kortste extra tijd (19 minuten) nodig voor de postbode.
We bekomen de kortste route (in tijd) waarbij elke straat minstens eenmaal doorlopen wordt door extra bogen PA, AB, CE, ED aan de graaf toe te voegen.



Opdracht: noteer het traject dat de postbode moet afleggen en bereken hoeveel tijd hij daar voor nodig heeft.

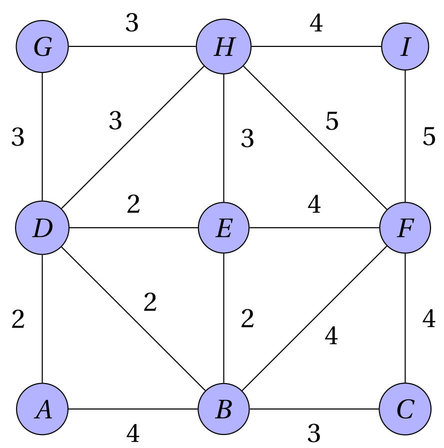
6.4 Algoritme voor het oplossen van het postbodeprobleem

Om het postbodeprobleem op te lossen combineren we de theorie over eulergrafen (zie module 1) met een kortste-pad-algoritme (bv. algoritme van Dijkstra, zie module 5).

1. Zoek alle knopen van oneven graad.
2. Bepaal voor elk tweetal knopen van oneven graad de lengte van het kortste pad. Deze kortste paden worden eventueel met het algoritme van Dijkstra berekend.
3. Bepaal alle mogelijke manieren waarop de knopen met oneven graad kunnen worden opgesplitst in paren.
4. Kies een opsplitsing met de kortste lengte. Voeg de bijhorende bogen toe aan de graaf.
5. De nieuwe graaf heeft alleen knopen met even graad. Zoek een euleriaans circuit in de graaf.

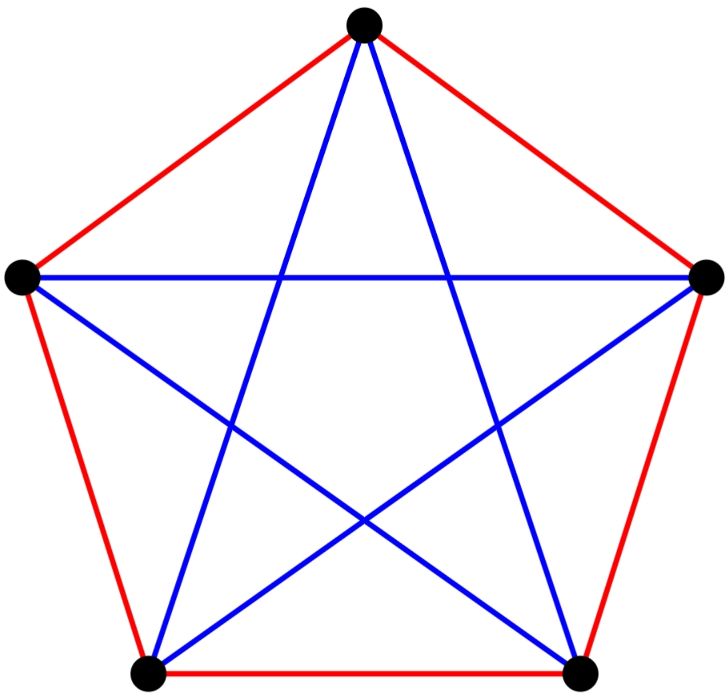
6.5 Oefening

In onderstaande wijk (weergegeven door een gewogen graaf) moet de post worden bezocht. De gewichten stellen de lengtes van de straten voor in km. De postbode mag beginnen waar hij wil, maar moet op dezelfde plek eindigen. Wat is de minimale afstand die de postpode moet afleggen om in de hele wijk de post te bezorgen? Welke route met hij dan lopen?



Module 7

Uitbreiding: feesten, vrienden en vreemden



7.1 Feestelijke problemen

Stel – je bevindt je op een kleinschalig feest met slechts 6 aanwezigen, jezelf inclusief. We gaan ervan uit dat elke twee personen op dat feest elkaar ofwel kennen ofwel niet kennen (in formele taal: we veronderstellen dat als persoon A persoon B kent, persoon B ook persoon A kent, m.a.w. deze relatie is symmetrisch).

Probleemstelling 1

Bewijs dat er onder de 6 aanwezigen altijd 3 personen bestaan die

- ofwel elkaar onderling kennen,
- ofwel elkaar onderling niet kennen.

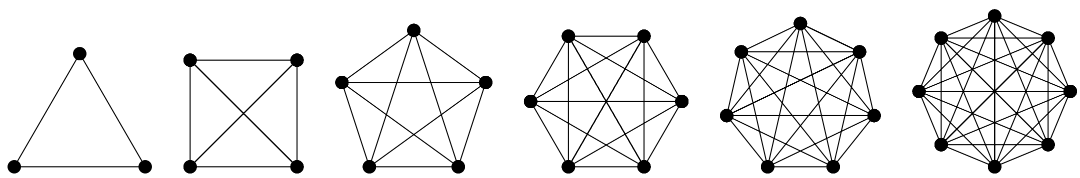
Bovenstaand probleem is een gekend **party problem**, zo bestaan er nog talloze andere. Wiskundigen hebben deze problemen geabstraheerd en gebundeld onder de noemer **Ramsey-theorie**, genoemd naar de Britse wiskundige en filosoof Frank P. Ramsey.[1]

Definitie

Een graaf $G' = (V', E')$ is een **deelgraaf** van een graaf $G = (V, E)$ als $V' \subseteq V$ en $E' \subseteq E$, met de afspraak dat E' enkel bestaat uit bogen die twee knopen uit V' verbinden.

Definitie

Een **complete graaf** is een enkelvoudige graaf waarvan elke twee knopen verbonden zijn door een boog.



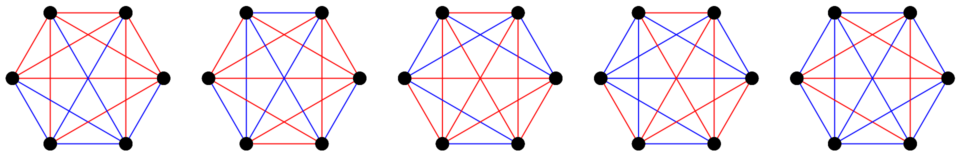
Een complete graaf bevat over het algemeen veel bogen (reken zelf uit hoeveel precies). Een feest zullen we aan de hand van dergelijke graaf voorstellen. Omdat we echter graag duidelijk willen maken wanneer twee knopen elkaar al dan niet kennen, kleuren we de bogen: blauw als de knopen elkaar kennen, rood als de knopen elkaar niet kennen.

[1] Als inleider tot deze module kan je de eerste tien minuten bekijken van de uiteenzetting van professor David Eelbode voor de Universiteit van Vlaanderen: `https://www.youtube.com/watch?v=xDPL6Q_D0ss`. Na het behandelen van de module kun je gerust de rest van de video bekijken.

Definitie

Een **bichrome graaf** is een graaf waarbij elke boog een kleur krijgt, gekozen uit een verzameling van twee kleuren (bv. blauw en rood). Een **monochrome graaf** is een graaf waarbij alle bogen dezelfde kleur hebben.

Het eerderbesproken feest met 6 aanwezigen wordt zo gemodelleerd aan de hand van een **bichrome complete graaf**; volgende grafen stellen telkens verschillende, mogelijke situaties voor.



Aan de hand van deze definities kunnen we probleemstelling 1 herformuleren als volgt.

Probleemstelling 2

Bewijs dat elke bichrome complete graaf van orde 6 altijd een monochrome complete deelgraaf bezit van orde 3.

Met andere woorden, kun je in eender welke bichrome complete graaf van orde 6 altijd een blauwe of rode driehoek terugvinden? Je kunt voor jezelf nagaan dat dit altijd het geval is bij bovenstaande voorbeelden!

Stelling 1

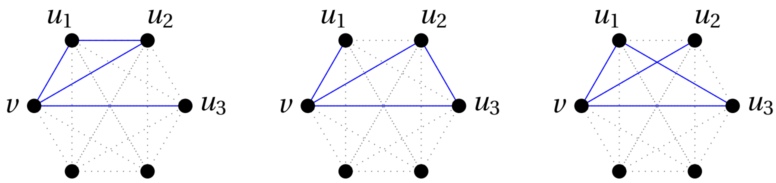
Elke bichrome complete graaf van orde 6 bezit een monochrome complete deelgraaf van orde 3.

Bewijs. Beschouw een bichrome complete graaf G van orde 6 en noem één van haar knopen v. Deze knoop heeft graad 5, en elke boog die deze knoop bevat is blauw of rood. Dit betekent dat er altijd minstens 3 van deze 5 bogen dezelfde kleur hebben. Zonder verlies van algemeenheid mogen we veronderstellen dat v bevat is in 3 blauwe bogen. Noem de andere uiteinden van deze 3 blauwe bogen u_1, u_2 en u_3. Er ontstaan twee mogelijke scenario's:

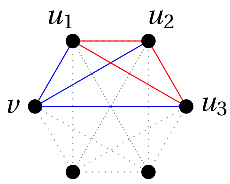
1. ofwel is één van de bogen uit $\{u_1u_2, u_2u_3, u_1u_3\}$ blauw,

2. ofwel zijn alle drie de bogen uit $\{u_1u_2, u_2u_3, u_1u_3\}$ rood.

Het eerste scenario leidt tot de volgende drie mogelijke situaties, afhankelijk van welke boog blauw blijkt te zijn; in elk van de situaties kunnen we een blauwe driehoek terugvinden:



Bij het tweede scenario ontstaat onmiddellijk een rode driehoek:



□

7.2 Ramseygetallen: orde in de chaos

Uit het vorige onderdeel hebben we bewezen dat eender welke bichrome complete graaf van orde 6 **altijd** een monochrome driehoek bezit. We kunnen ons echter afvragen of deze eigenschap nog steeds geldt bij bichrome complete grafen van een lagere orde, bijvoorbeeld orde 5?

Probleemstelling 3
Bezit elke bichrome complete graaf van orde 5 een monochrome driehoek? Indien ja, bewijs; indien nee, geef een tegenvoorbeeld.

Het antwoord op bovenstaande vraag is negatief. Hierdoor kunnen we het volgende vaststellen: *vanaf een bichrome complete graaf op z'n minst* 6 *knopen bezit, bestaat er altijd een monochrome driehoek.*

Dit geldt voor elke bichrome graaf, waar er over het algemeen zeer veel van zijn (van orde 6 bestaan er maar liefst $2^{15} = 32768$ bichrome complete grafen[2]). Deze veelheid aan mogelijke situaties wordt vaak geassocieerd aan **chaos**, terwijl het vinden van een monochrome complete deelgraaf gepaard gaat met het vinden van **orde**. Hoe groter de orde is van een bichrome graaf, hoe groter de kans is voor het vinden van monochrome complete deelgrafen. Daarom dat Ramseytheorie vaak informeel beschreven wordt als de vraag: **kan er altijd orde gevonden worden in chaos?**

[2] Hier speelden echter een beetje vals, omdat we álle bichrome complete grafen van orde 6 telden, dus ook diegene die na een rotatie/spiegeling/kleurenwisseling/... in essentie dezelfde graaf zijn. De grootteorde van dit aantal zal echter nog steeds exponentieel stijgen ten opzichte van de orde van de graaf!

Ramseytheorie slaat op het vinden van **Ramseygetallen**; informeel gesproken zijn dit getallen die de grens aanduiden vanaf waar er orde optreedt binnen chaos.

Definitie

Neem aan dat b en r natuurlijke getallen zijn. Het Ramseygetal $R(b,r)$ is het unieke natuurlijke getal met de volgende twee eigenschappen:

1. Er bestaat een bichrome complete graaf met $R(b,r)-1$ knopen die noch een blauwe complete deelgraaf van orde b bezit, noch een rode complete deelgraaf van orde r.

2. Elke bichrome complete graaf met $R(b,r)$ knopen bezit oftewel een blauwe complete deelgraaf van orde b, oftewel een rode complete deelgraaf van orde r.

Deze getallen werden genoemd naar de wiskundige Frank Ramsey aangezien hij als eerste bewees dat zulke getallen altijd bestaan, ondanks we niet altijd weten wat hun precieze waarde is. Dat klinkt misschien wat gek, maar in essentie bewees Ramsey dat het getal $R(b,r)$ begrensd moet zijn, en dus niet oneindig groot kan worden.

Sommige waarden zijn echter wel gekend.

Stelling

$R(3,3)=6$.

Bewijs. Het tegenvoorbeeld dat werd gevonden bij probleemstelling 3 is een bichrome complete graaf met 5 knopen die noch een blauwe complete deelgraaf van orde 3 bezit, noch een rode complete deelgraaf van orde 3.
Stelling 1 bewijst dat elke bichrome complete graaf met 6 knopen ofwel een blauwe complete deelgraaf van orde 3 bezit, ofwel een rode complete deelgraaf van orde 3. □

Probleemstelling

Bewijs, voor elke $n \in \mathbb{N} \setminus \{0,1\}$, dat $R(2,n) = R(n,2) = n$.

Hierrond wordt tot op de dag van vandaag nog steeds onderzoek gedaan. Zo hebben de wiskundigen Evans, Pulham en Sheehan reeds in 1979 bewezen dat $R(4,4) = 18$, terwijl de wiskundigen Brendan McKay en Stanisław Radziszowski in 1995 bewezen dat $R(4,5) = 25$.
Er zijn echter nog steeds (erg veel) waarden die nog niet gekend zijn. Zo kennen we de waarde van $R(5,5)$ **niet**; het enige wat we wél weten is dat $43 \leqslant R(5,5) \leqslant 48$. Sommige wiskundigen hebben goede argumenten om te vermoeden dat de ware waarde van $R(5,5)$ gelijk is aan 43,

maar zekerheid hebben we (nog) niet.

Hoe groter de parameters b en r worden van $R(b,r)$, hoe minder goed we de waarde kunnen afschatten. Zo weten we dat $102 \leqslant R(6,6) \leqslant 165$, $205 \leqslant R(7,7) \leqslant 540$ en $282 \leqslant R(8,8) \leqslant 1870$. Niet enkel worden deze afschattingen alsmaar slechter, ook vermoedt de meerderheid van wiskundigen dat we binnen de eerstvolgende eeuwen weinig verbetering kunnen verwachten, simpelweg omdat de huidige computationele rekenkracht voorlopig ontoereikend is. Zo bestaat het volgende befaamde citaat van Hongaars wiskundige Paul Erdős, vrij vertaald naar het Nederlands:
"Beeld je een alienleger in, vele malen krachtiger dan de onze, die landt op Aarde en ons de exacte waarde van $R(5,5)$ *eist, anders vernietigen ze onze planeet. In dat geval verzamelen we best onze krachtigste supercomputers en slimste wiskundigen om te pogen deze waarde te vinden. Stel echter, in de plaats, dat ze vragen naar de waarde* $R(6,6)$. *In dat geval verzamelen we best onze pienterste ingenieurs en dapperste militairen en gaan we in de tegenaanval."*

Dank

We danken prof. Hendrik Van Maldeghem voor het nalezen van deze syllabus en voor de feedback.
We danken ook Gommaar Maes (collega onderwijsbegeleider EDUMA wiskunde UGent en leerkracht GO!Atheneum Mariakerke) voor het ter beschikking stellen van het deel grafen uit zijn cursus problem solving.

Bibliografie

[1] https://www.wiskunde.ugent.be/wiskunde-kiezen/over-wiskunde.php

[2] Veerle Fack en Hendrik Van Maldeghem, UGent, cursus grafentheorie (september 2013)

[3] Bart De Bruyn, UGent, cursus Discrete wiskunde II, (academiejaar 2019-2020)

[4] Gommaar Maes, GO!Atheneum Mariakerke, cursus problem solving (2021-2022)

[5] Hajo Broersma, Grafen in de praktijk, Epsilon Uitgaven, Zebrareeks deel 14

[6] Michel Roelens en Els Vanlommel, Redeneren en puzzelen met grafen, Uitwiskeling 36/2 (lente 2020)

[7] Alex Van den Brandhof, Postbodes, handelsreizigers en kortste paden, Pythagoras, 48ste jaargang nr 6 - juni 2009

[8] https://www.uitwiskeling.be/2020/01/26/de-zeven-bruggen-van-koningsbergen/?v=d3dcf429c679

[9] https://nl.wikipedia.org/wiki/Zeven_bruggen_van_Koningsbergen

[10] https://www.maa.org/press/periodicals/convergence/leonard-eulers-solution-to-the-konigsberg-bridge-problem

[11] https://commons.wikimedia.org/wiki/File:Solutio_problematis_ad_geometriam_situs_pertinentis,_Fig._1_-_Cleaned_Up.png

[12] https://lirias.kuleuven.be/retrieve/241940

[13] https://homepages.cwi.nl/~lex/files/graphs1_3.pdf

[14] https://www.ru.nl/publish/pages/959798/voorbereidingsmateriaal.pdf

[15] https://ocw.tudelft.nl/wp-content/uploads/Optimaliseren_in_Netwerken_Module1.pdf

[16] https://nl.wikipedia.org/wiki/Paardenrondgang

[17] https://www.archimedes-lab.org/knight_tour.html

[18] https://en.wikipedia.org/wiki/Three_utilities_problem

[19] http://www.wiskundemeisjes.nl/20090420/ode-aan-dijkstra/

[20] https://www.program-uurtje.org/kortstepad.html

[21] https://maken.wikiwijs.nl/userfiles/e5aa4e5c68c7507ece2913f7b184c3cbccf85cd8.pdf

[22] https://www.universiteitvanvlaanderen.be/college/hoe-kan-je-met-eenvoudige-wiskunde-een-computer-overtreffen