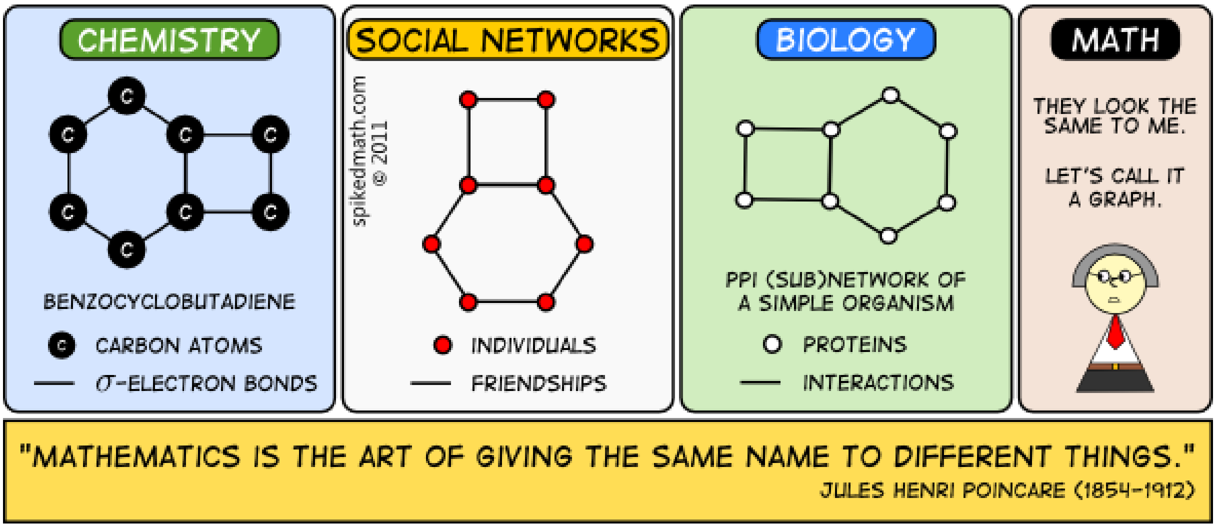
Grafen

Minimale opspannende bomen

onderzoekend leren



Anneleen De Schepper
Vakdidactiek Wiskunde II
2021-2022

Opfrissing

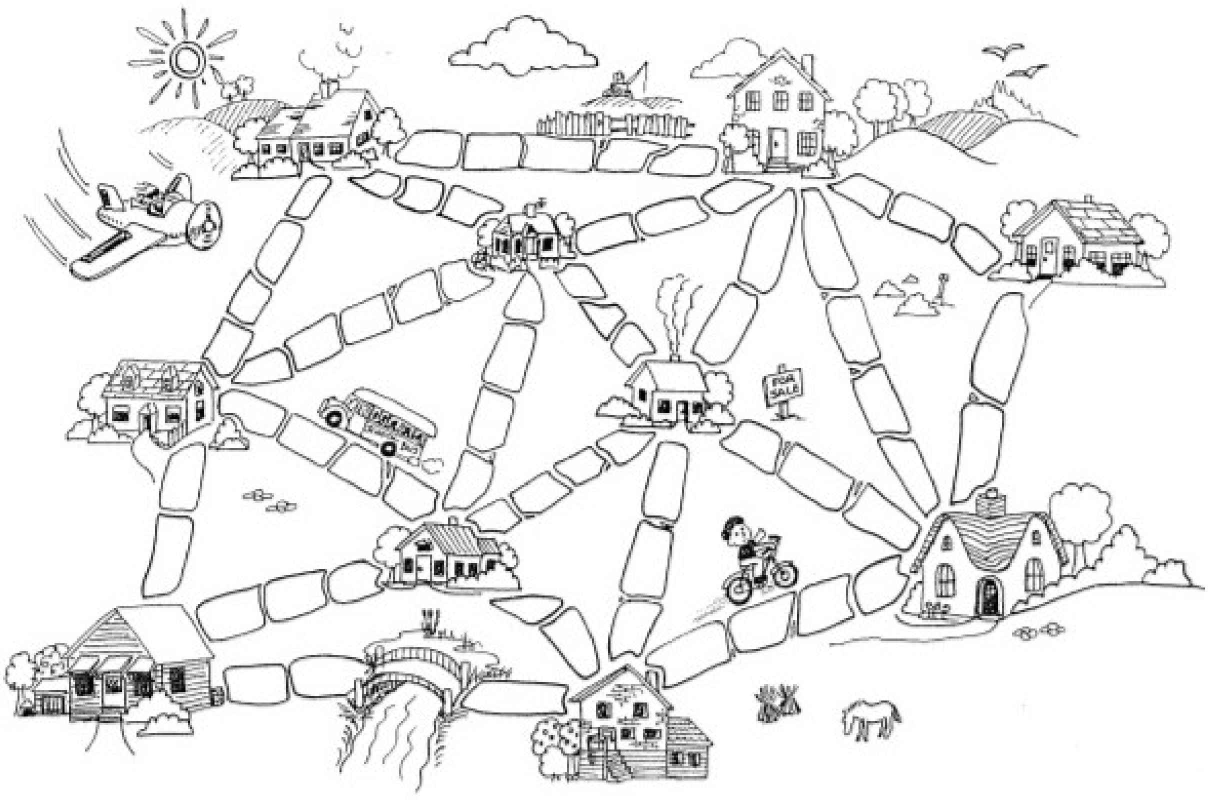
Zij $G = (V, E)$ een graaf met V als knopenverzameling en E als bogenverzameling. We gaan er in deze bundel van uit dat de volgende begrippen reeds gekend zijn, we geven de definities ter opfrissing.

Definities

- De **orde** van G is het aantal knopen, de **grootte** van G is het aantal bogen.
- Twee knopen van G die verbonden worden door een boog, worden **naburig** genoemd. De **graad** van een knoop van G is het aantal buren van die knoop.
- Een **wandeling** in G is een rij van aaneensluitende bogen, d.w.z., twee opeenvolgende bogen hebben steeds een uiteinde gemeen.
- We noemen deze wandeling een **pad** als in elke knoop ervan ten hoogste twee bogen van de wandeling samenkomen.
- Een pad dat vertrekt en eindigt in dezelfde knoop is een **cykel**.
- We noemen G **samenhangend** als er voor elk paar knopen een wandeling is van de ene naar de andere knoop.
- 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.

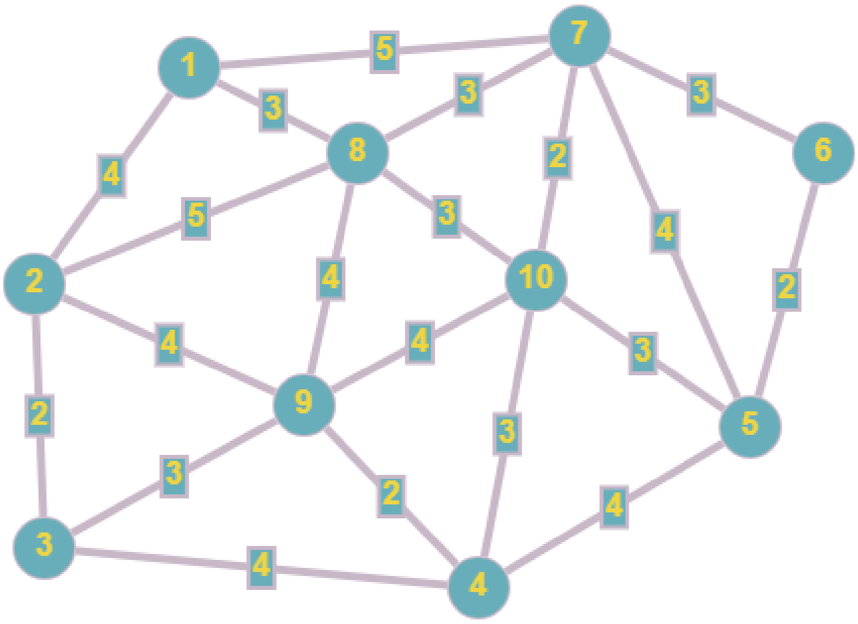
1 Inleidend voorbeeld

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. Aan jou om het bedrijf te helpen bepalen hoe het zijn kabels moet leggen.



Opdracht 1.1 Ga naar `https://graphonline.ru/nl/`. Stel bovenstaand netwerk voor als gewogen een graaf G: elke knoop stelt een huis voor, en twee huizen zijn verbonden door een boog als er een stenen weg tussen hen loopt (het brugje telt mee als steen) met als gewicht het aantal stenen dat deze weg telt.

Opdracht 1.2 Zoek in de graaf hieronder een mogelijk kabelnetwerk en bepaal de som van de labels van de gebruikte bogen. Controleer ook of je dezelfde gewogen graaf als hieronder had getekend.



Opdracht 1.3 Klik op de website `https://graphonline.ru/nl/`. op 'algoritmes' en dan op 'minimale opspannende boom'. Vergelijk het resultaat met jouw netwerk van hierboven.

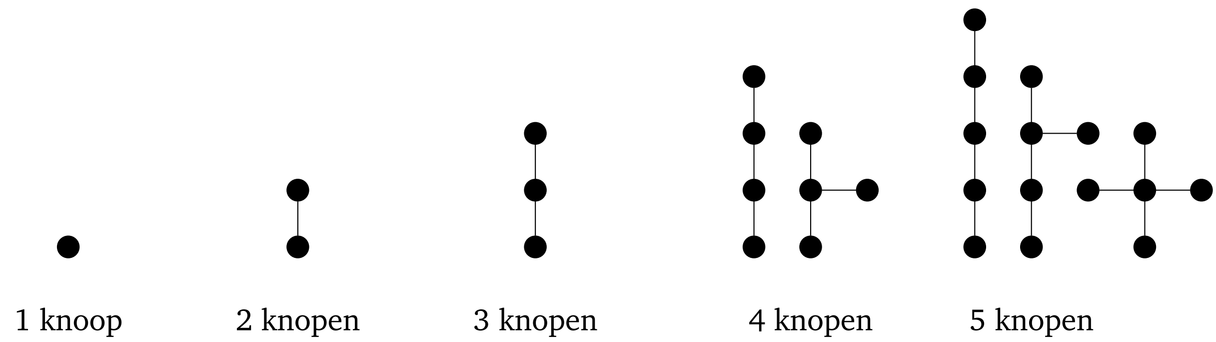
Opdracht 1.4 Het netwerk dat je opstelt is zelf ook een graaf. Geef twee eigenschappen van deze graaf.

We zullen aanleren wat de computer doet om zo een minimaal netwerk te vinden. Hiervoor bespreken we eerst in wat meer detail het soort grafen dat voorkomt als zo een netwerk.

2 Bomen: definitie en eigenschappen

De grafen die we zonet ontmoetten, zijn wat we noemen **bomen**. Ze hebben tal van praktische toepassingen, zoals reeds geïllustreerd, maar ook in een meer theoretische context zijn ze belangrijk. Reden genoeg om er ons in te verdiepen.

Definitie 2.1 Een samenhangende graaf zonder cykels wordt een **boom** genoemd.



Figuur 1: Alle bomen met ten hoogste 5 knopen.

Opdracht 2.1 Teken alle bomen met 6 knopen (*hint: zo zijn er 6*).

Opdracht 2.2 Alle bomen met 6 knopen hebben 5 bogen (tel maar na). Kan je een graaf $G = (V, E)$ vinden met 6 knopen in elk van de volgende gevallen? Zo ja, geef een graaf.

aantal bogen	G is samenhangend	G heeft geen cykels
< 5		
> 5		
$= 5$		

We veralgemenen nu de bevindingen van het voorbeeld van een graaf met 6 knopen naar een graaf met n knopen.

Stelling

Zij G een graaf met n knopen. Dan is G een boom **als en slechts** als G $n-1$ bogen heeft.

Als G samenhangend is en meer dan $n-1$ bogen heeft, **dan** bevat G een cykel.

Als G geen cykels bevat en minder dan $n-1$ bogen heeft, **dan** is G niet samenhangend.

3 Opspannende bomen

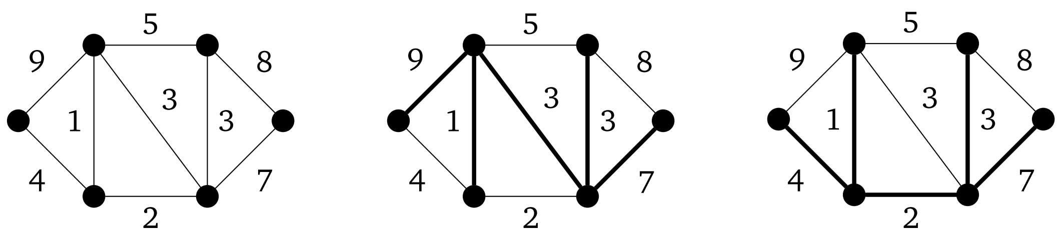
3.1 Minimale opspannende bomen en gewogen grafen

We zagen reeds dat het netwerk dat we zouden voorstellen aan het bedrijf, een voorbeeld zal zijn van een boom. Dit is niet zomaar eender welke boom, we vragen hierbij nog twee extra eigenschappen: elke knoop van de graaf moet erin bevat zijn, en de som van de gewichten van de gebruikte bogen moet minimaal zijn. We gieten dit in de volgende definitie:

Definitie

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

Hieronder vind je een voorbeeld.



Figuur 2: Van links naar rechts: Een gewogen graaf G, een opspannende boom H van G met $g(H) = 23$, een mimale opspannende boom H_0 met $g(H_0) = 17$.

We geven je nu twee methodes, die gegarandeerd werken, om in een gewogen graaf een minimale opspannende boom te vinden. Zo een methode die altijd werkt, noemen we een **algoritme**. Typisch start een algoritme 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 – althans, dat is de bedoeling! Laat ons dit even bekijken.

3.2 Algoritme van Kruskal

We construeren stap voor stap een opspannende boom, startende van één boog met minimaal gewicht en dan steeds bogen toevoegend hieraan die geen cykels doen ontstaan.

Algoritme van Kruskal

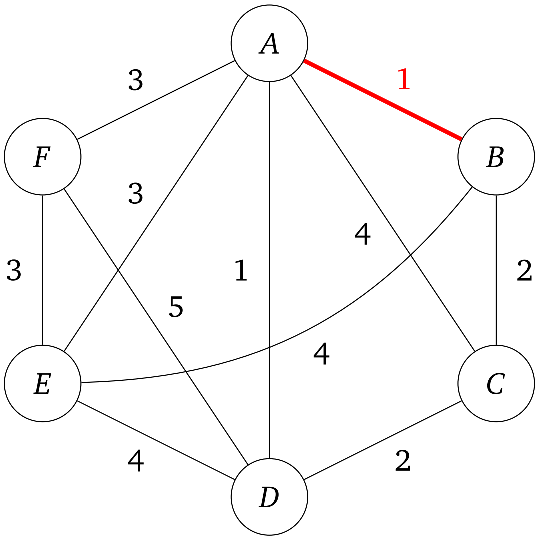
Ga uit van een samenhangende gewogen graaf G met minstens 3 knopen en een graaf H met dezelfde knopen als G, maar nog geen bogen.

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.

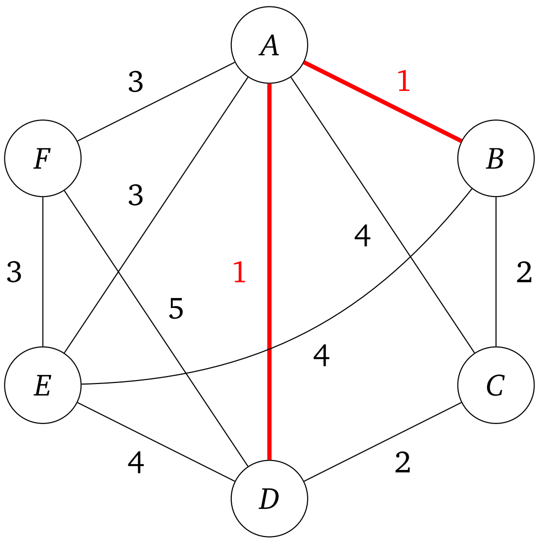
   Voeg deze boog toe aan H.
3. **Einde**: stop op het moment dat er bij stap 2 geen nieuwe boog gekozen kan worden. H is dan de minimaal opspannende boom van G.

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

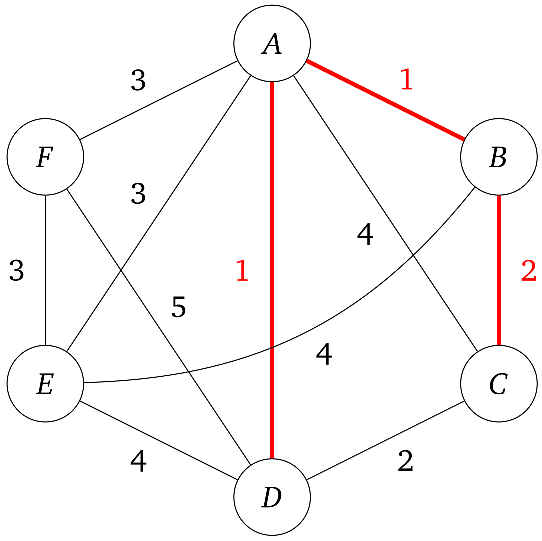
We vertrekken van een gegeven graaf G en we kiezen een boog met minimaal gewicht, bijvoorbeeld boog AB met gewicht 1 (we hadden ook kunnen kiezen voor boog AD).



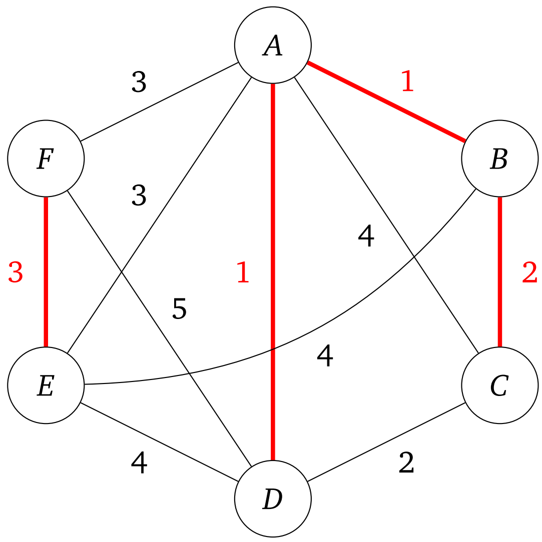
Van de overblijvende bogen van G is het minimale voorkomende gewicht nog steeds 1. Nu is AD de enige boog met dat gewicht die nog voorkomt. Boog AD toevoegen aan H doet geen cykel ontstaan. We voegen dus boog AD toe.



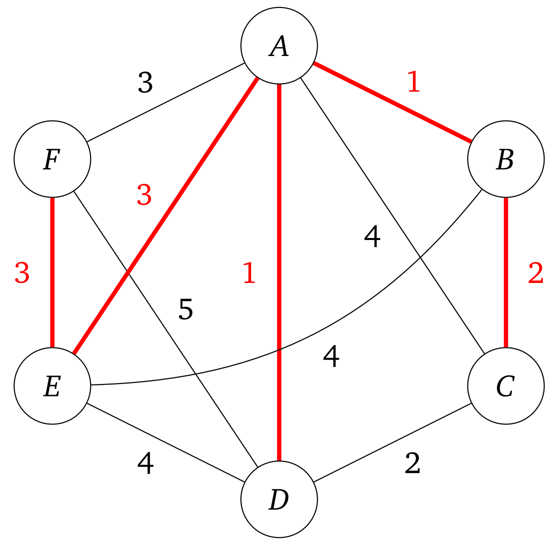
Het minimale gewicht dat nu voorkomt onder de overblijvende bogen is 2. Er zijn twee bogen met gewicht 2, en toevoegen van één ervan zorgt niet voor cykels in H. We kiezen om boog BC toe te voegen (boog CD had echter ook gekund).



Boog CD heeft minimaal gewicht (2) maar kunnen we nu niet toevoegen aan H omdat we dan een cykel krijgen. We kiezen dus een boog van gewicht 3, bijvoorbeeld EF, om toe te voegen aan H (boog AF of boog AE had ook gekund).



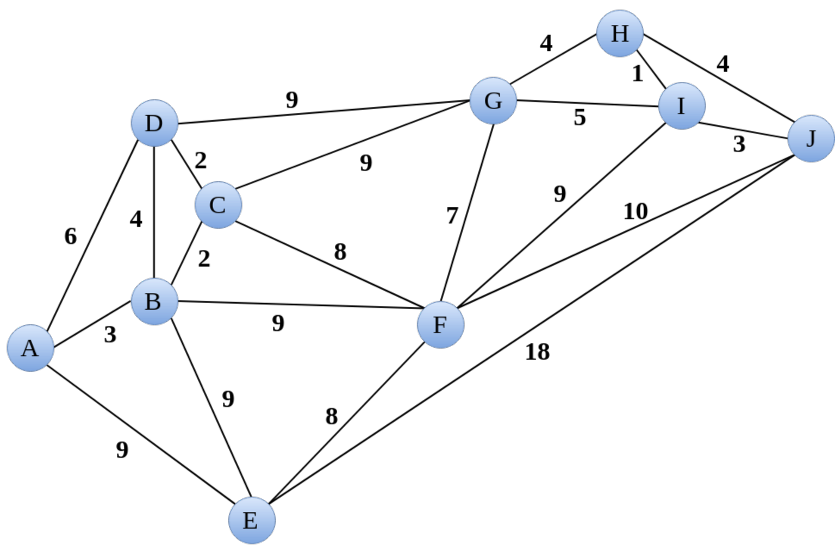
We kunnen nu kiezen uit bogen AE en bogen AF: beide bogen hebben minimaal gewicht en zorgen niet voor een cykel in H. We kiezen ervoor om AE toe te voegen.



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

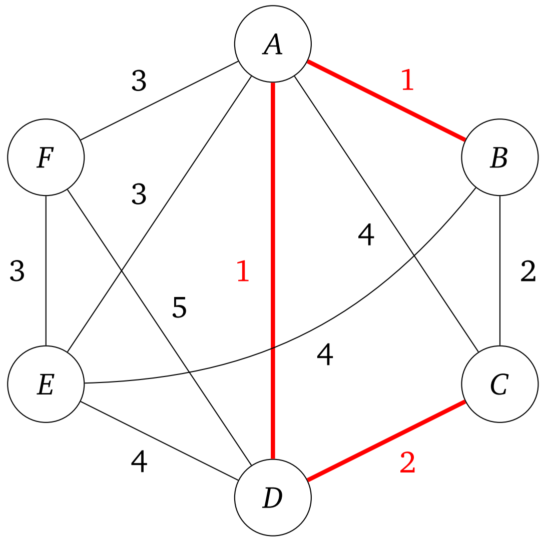
Opdracht 3.1 Ga naar `https://www.geogebra.org/m/qCetczUH` en klik op 'Algoritme van Kruskal'. Doorloop het voorbeeld stap voor stap en probeer zelf steeds eerst een geschikte boog te vinden.

Opdracht 3.2 Pas het algoritme van Kruskal toe op onderstaande voorbeeld.
Vergelijk je resultaat met de oplossing die je kan vinden m.b.v. `https://graphonline.ru/nl/`. Klik op `http://graphonline.ru/nl/?graph=SdzsYjDfPPlEoEFn`. Zoek naar de minimaal opspannende boom.
Zijn de bomen gelijk? En de gewichten?

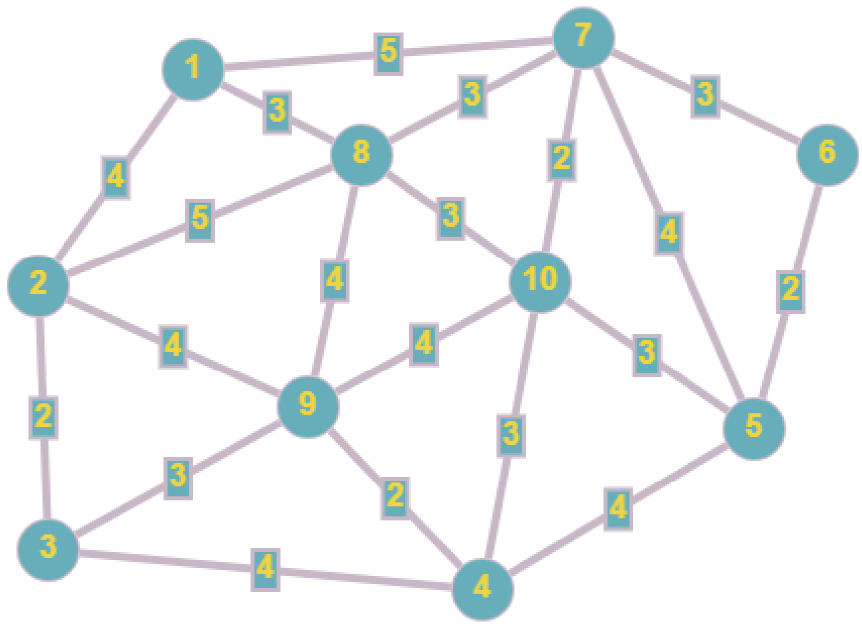


Doordenker. Zou het kunnen dat G nog een andere opspannende boom bevat met een even laag of zelfs lager gewicht?

Opdracht 3.3 Pas het algoritme van Kruskal toe op de graaf in het voorbeeld, maar gebruik in stap drie boog CD in plaats van boog BC. Vergelijk ook hier de bomen en hun gewichten. Wat zijn je bevindingen?



Opdracht 3.4 Pas het algoritme van Kruskal toe op de graaf uit het inleidend voorbeeld.



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: het resultaat is in elk geval een minimaal opspannende boom. Dit betekent echter niet dat dit de enige beste oplossing is, er kunnen nog andere opspannende bomen zijn met een even laag gewicht.

3.3 Algoritme van Prim

We bestuderen nog een tweede algoritme om een minimale opspannende boom stap voor stap te construeren. Ditmaal kiezen we onze opeenvolgende bogen zodanig dat de deelgraaf die ontstaat op elk moment een boom is (en dus ook op elk moment samenhangend is, in tegenstelling tot de situatie bij het algoritme van Kruskal).

We zullen op elk moment van het stappenplan een verzameling N bijhouden die de knopen bevat die tot dan toe tot de boom in opbouw behoren.

Algoritme van Prim

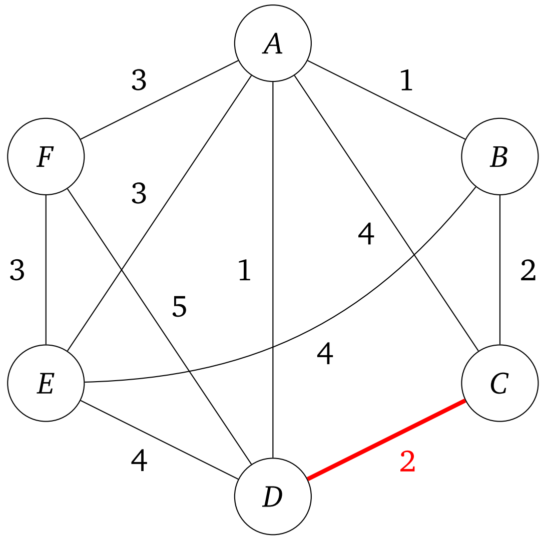
Ga uit van een samenhangende gewogen graaf $G = (V, E)$ met minstens 3 knopen.

1. **Begin**: kies een willekeurige knoop a.
   Stel $N = \{a\}$.
2. **Herhaal** volgende stap: Kies een boog van G zodat
   - één uiteinde van de boog tot N behoort en het ander uiteinde niet
   - het gewicht van die boog zo laag mogelijk is.

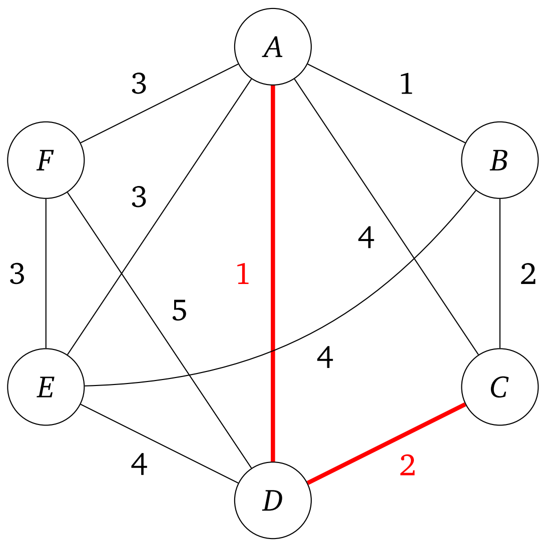
   Voeg het uiteinde van deze boog dat niét in N zit toe aan N.
3. **Einde**: stop als alle knopen van G tot N behoren.

Net zoals het algoritme van Kruskal levert het algoritme van Prim in elk geval een minimaal opspannende boom op. Hieronder zie je het algoritme toegepast op een voorbeeld.

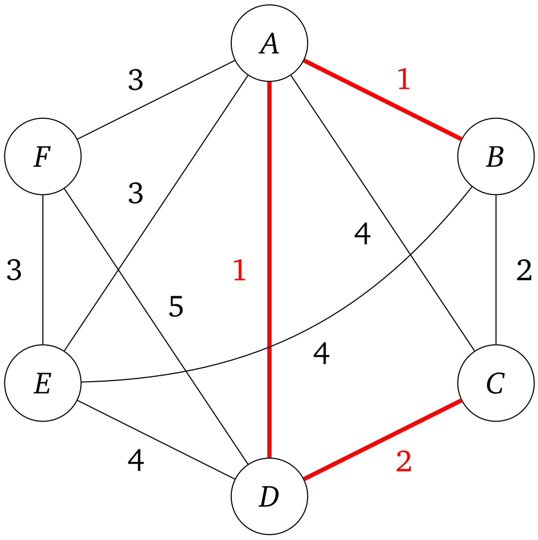
In de beginstap vertrekken we vanuit een willekeurige knoop van een gegeven graaf G: we kiezen C en stellen dus $N = \{C\}$. Vervolgens bekijken we de bogen die vertrekken uit C. Het laagste gewicht is 2 en er zijn twee zulke bogen BC en CD. We kiezen boog CD. We voegen D toe aan N en krijgen $N = \{C, D\}$.



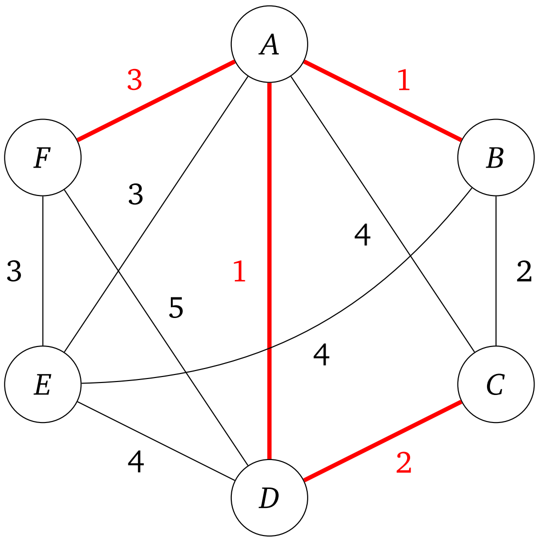
We bekijken nu alle bogen die vertrekken uit C of D en die nog niet gekozen zijn. Boog DA heeft van al deze bogen het kleinste gewicht (namelijk 1), dus voegen we boog DA toe aan de deelgraaf. We krijgen $N = \{C, D, A\}$.



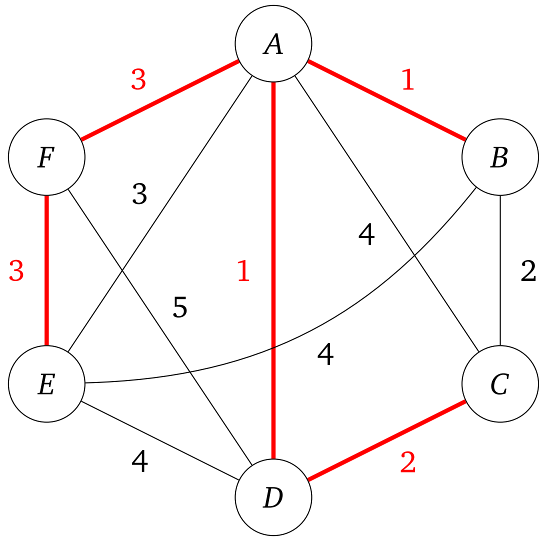
Boog AB is de boog met het kleinste gewicht als we alle bogen beschouwen die één uiteinde in $N = \{C, D, A\}$ hebben. We voegen dus boog AB toe aan onze boom en knoop B aan N, dus N wordt nu $\{C, D, A, B\}$.



Bogen AF en AE zijn de bogen met het kleinste gewicht (zijnde 3) als we alle bogen met één uiteinde in $N = \{C, D, A, B\}$ elkaar vergelijken. We kiezen één van deze twee bogen, bijvoorbeeld AF, en voegen deze boog toe aan de boom. We stellen $N = \{C, D, A, B, F\}$.



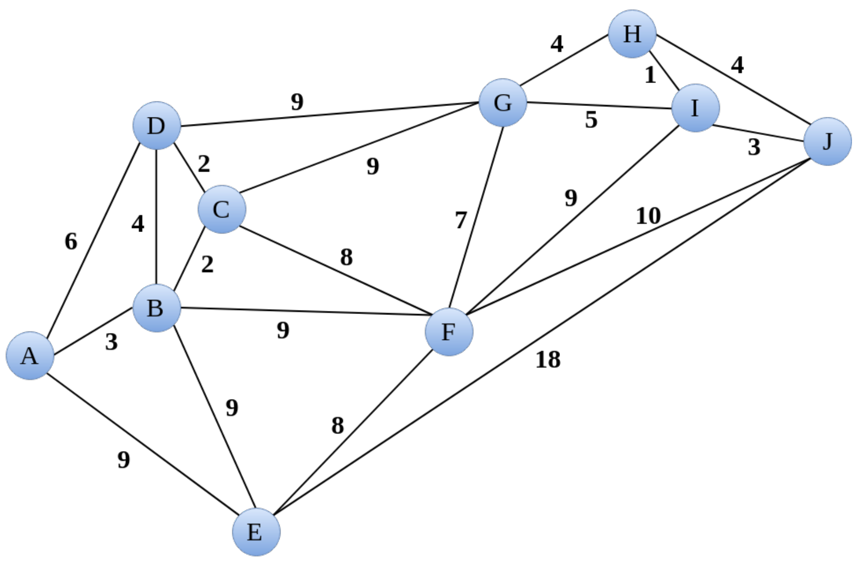
De twee bogen die we nu kunnen kiezen zijn FE en AE. We kiezen FE en stellen $N = \{C, D, A, B, F, E\}$.



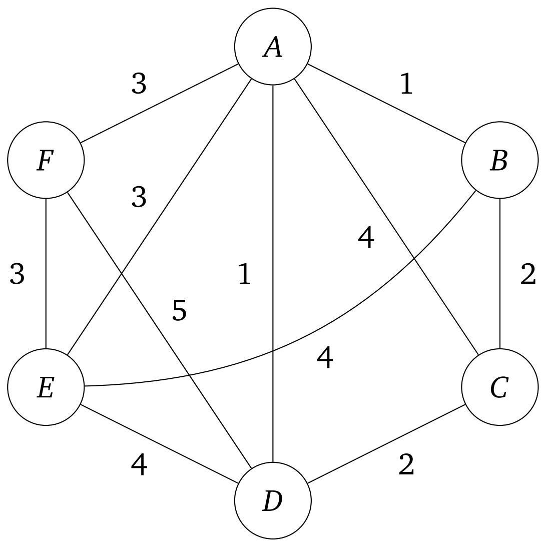
Alle knopen van G behoren tot de deelgraaf. We hebben een minimaal opspannende boom van G gevonden. Het gewicht van deze deelgraaf is 10.

Opdracht 3.5 Ga naar `https://www.geogebra.org/m/qCetczUH` en klik op 'Algoritme van Prim'. Doorloop het voorbeeld stap voor stap en probeer zelf steeds eerst een geschikte boog te vinden.

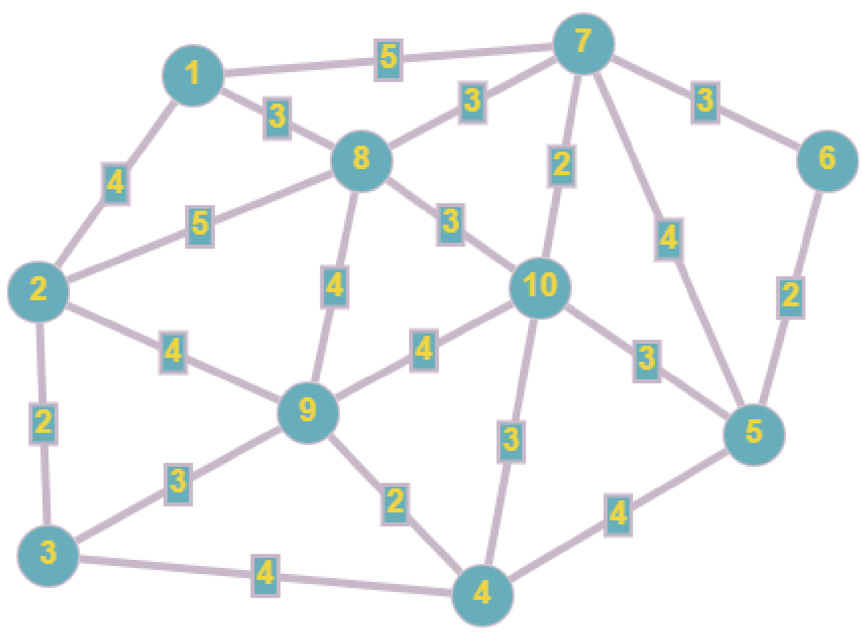
Opdracht 3.6 Pas het algoritme van Prim toe op de graaf uit oefening 3.2 (hieronder nog eens gegeven) en vergelijk ook hier je resultaten met behulp van `http://graphonline.ru/nl/?graph=SdzsYjDfPPlEoEFn`.



Opdracht 3.7 Doe hetzelfde voor de graaf in het voorbeeld hierboven (hieronder nog eens gegeven), maar start met een andere knoop (naar keuze). Wat zijn je bevindingen?



Opdracht 3.8 Pas het algoritme van Prim toe op de graaf uit het inleidend voorbeeld.



Referenties

[1] John Clark, Derek Alan Holton, *A first look at graph theory*, World Scientific Publishing Company, 1991 (online beschikbaar op https://inoerofik.files.wordpress.com/2014/11/firstlook_graphtheory.pdf)

[2] Hendrik Van Maldeghem, *cursusnota's Grafentheorie 2013-2014*, UGent.

[3] Robin J. Wilson, *Introduction to graph theory (fifth edition)*, Pearson Education Limited, 2010.

[4] https://personal.utdallas.edu/~besp/teaching/mst-applications.pdf

[5] https://visualgo.net/en/mst