A továbbiakban ismertetünk egy módszert, amely lehetővé teszi a cimkézett fák megadását a megfelelő ábra nélkül, az ún. *Prüfer-kód* segítségével. A fa persze megadható az élek felsorolásával vagy a szomszédsági mátrixszal is, de ez egy tömörebb reprezentáció, amely ugyanakkor elvezet a Cayley-tétel bizonyításához is.

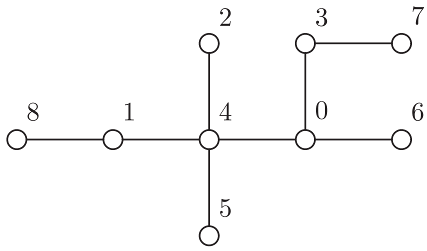
Tekintsünk egy n pontú cimkézett gráfot, a cimkék legyenek a $0, 1, 2, ..., n-1$ számok. Akkor a fa Prüfer-kódja egy, a $0, 1, 2, ..., n-1$ számok közül $(n-2)$ számból álló $\boxed{k_1\, k_2 \ldots k_{n-2}}$ alakú kód, ahol fontos a számok sorrendje.

A 0 cimkéjű pontot a **fa gyökeré**nek nevezzük, ez a fa tetszőleges rögzített pontja lehet. A gyökértől különböző elsőfokú pontokat a **fa levelei**nek nevezzük. Biztosan van legalább egy levél. Mi több, ha a gyökér nem elsőfokú, akkor létezik legalább két levél, lásd 19.6. Tétel.

A Prüfer-kódot a következőképpen kapjuk meg:

Megkeressük a levelek közül a legkisebb cimkéjűt (t_1) és leírjuk ennek a szomszédját (k_1). Ez lesz a kód első száma (k_1). Töröljük a t_1 pontot a hozzátartozó éllel együtt. Megkeressük a megmaradt fa levelei közül a legkisebb cimkéjűt (t_2) és leírjuk ennek a szomszédját (k_2). Ez lesz a kód második száma (k_2). Most töröljük a t_2 pontot a hozzátartozó éllel együtt. Keressük meg a megmaradt fa levelei közül a legkisebb cimkéjűt (t_3) és leírjuk ennek a szomszédját (k_3). Ezt ismételjük (sorra letépjük a leveleket) mindaddig, amíg egy két pontból álló fa nem marad. Így összesen $(n-2)$ pontot töröltünk, a törölt pontok szomszédaiból álló kód tehát valóban $(n-2)$ számból áll.

Például, adott a 24. ábrán levő $n = 9$ pontú fa. Először a 2 pontot kell törölni: $t_1 = 2$, ennek szomszédja a $k_1 = 4$, töröljük az 5 pontot: $t_2 = 5$, ennek a szomszédja a $k_2 = 4$, tovább $t_3 = 6$ és $k_3 = 0$, $t_4 = 7$ és $k_4 = 3$, $t_5 = 3$ és $k_5 = 0$, $t_6 = 8$ és $k_6 = 1$, $t_7 = 1$ és $k_7 = 4$ (rendre letépjük a megmaradt fa leveleit). Készítsünk egy táblázatot így:



24. ábra

2	5	6	7	3	8	1
4	4	0	3	0	1	4

A kód tehát ennek a második sora: $\boxed{4\,4\,0\,3\,0\,1\,4}$. Megjegyezzük, hogy így végül megmarad a $4-0$ él, a 4 törlésével a 0-t is leírhatnánk, de ez felesleges, mert mindig a 0 gyökér az utolsó szám.

Ebben az eljásban az az igazán figyelemre méltó, hogy egy ilyen kód megadásával rekonstruálható a fa. Tehát

<u>**19.9. Tétel.**</u> *A Prüfer-kód egyértelműen meghatározza a fát.*

Bizonyítás. Megmutatjuk, hogy a Prüfer-kód ismeretében egyértelműen megadható a fa $n-1$ éle. Tekintsük a következő példát. Adott az $\boxed{5\,2\,4\,1\,0\,2\,3\,1}$ kód. Ez 8 számból áll, ezért $n-2 = 8$, tehát egy $n = 10$ pontú fáról van szó. Melyek voltak vajon azok a $t_1, t_2, ..., t_8$ pontok, amelyek törlésével ezt kaptuk? Mi állhat az alábbi táblázat első sorában?

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
5	2	4	1	0	2	3	1

Lehet-e $t_1 = 1$? Nem, mert mert ekkor az első lépésben törölnünk kellett volna, és nem szerepelhetne a táblázat második sorában. Lehet-e $t_2 = 2$? Nem, ugyanezen oknál fogva. Így $t_1 \neq 1, 2, 3, 4, 5$. Következik innen, hogy $t_1 = 6$? Igen, mert ha a 6 nem lenne elsőfokú pont, akkor szomszédja lenne valaminek, és bekerült volna a második sorba. Tehát $t_1 = 6$.

Mi lehet t_2? Az első oszlop elhagyásával az 5 kikerül a második sorból, és az 5 lesz az a legkisebb szám, amely nem szerepel a második sorban. Tehát $t_2 = 5$. Tovább hasonlóan: $t_3 = 7$, $t_4 = 4$, $t_5 = 8$, $t_6 = 9$, $t_7 = 2$, $t_8 = 3$, azaz:

6	5	7	4	8	9	2	3
5	2	4	1	0	2	3	1

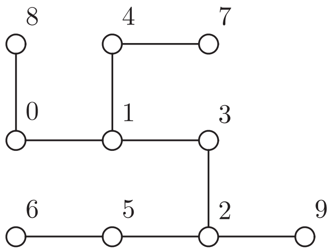
A szabály tehát ez: *az első sor minden eleme az a legkisebb pozitív szám, amely nem szerepel korábban az első sorban vagy később a második sorban.*

Hiányzik még az 1, ez alkotja a 0 gyökérrel az utolsó élet. Ezzel kiegészítve kapjuk a bővitett táblázatot:

6	5	7	4	8	9	2	3	1
5	2	4	1	0	2	3	1	0

Az oszlopok jelentik az éleket és ennek alapján rekonstruálható a fa. Kezdjük az utolsó oszloppal és haladjunk visszafelé, így egy egyre bővülő fát kapunk(!), lásd a 25. ábrát.

Ezek a meggondolások teljesen általánosak és a tulajdonság bizonyítását adják. □



25. ábra

19.10. Feladat. Melyik az a fa, amelynek a Prüfer-kódja $\boxed{5\,3\,2\,3\,4\,1\,7\,2\,0\,3}$?

Nem lehetne egyszerűbben kódolni? Felejtsük el egy rövid időre a Prüfer-kódot és tekintsük a következő feladatot:

19.11. Feladat. Próbálkozzunk a következő egyszerűbbnek tűnő kódolással: Adott egy cimkézett fa, ennek rögzített a gyökere. Vegyük sorra az éleket és írjuk le előbb a gyökértől távolabbi pontot, majd a gyökérhez közelebbi pontot. Az éleket rendezzük az elsőként leírt pontok növekvő sorrendjében. Pl. a 24. ábrából kindulva ezt kapjuk:

1	2	3	4	5	6	7	8
4	4	0	0	4	0	3	1

A próbakód legyen a táblázat második sora: 4 4 0 0 4 0 3 1. Egy n pontú fának így megfeleltetünk $n-1$ számot.

a) Biztos az, hogy az első sorban mindegyik szám egyszer szerepel?

b) Rekonstruálható-e ennek a próbakódnak alapján a fa? Vizsgáljuk a következő próbakódokat: 0 1 2 3 4 5 6 7 8, 8 7 6 5 4 3 2 1 0, 0 0 0 0 0 0, 3 1 2 5 4 5.

c) Miért nem jó ez a módszer? Miért nem lesz ez kód?

Térjünk vissza a Prüfer-kódhoz és igazoljuk a Cayley-tételt.

19.12. Tétel. (Cayley-tétel) *Az n pontú cimkézett fák száma* $\boxed{x_n = n^{n-2},\ n \geq 1}$.

Bizonyítás. Ahogy azt láttuk, a Prüfer-kód bijektív megfeleltetést létesít az n pontú cimkézett fák és azok között az $n-2$ tagú számsorozatok között, amelyek minden tagja a $0, 1, 2, ..., n-1$ számok valamelyike. Az ilyen sorozatok száma $\overline{V}_n^{n-2} = n^{n-2}$ (ismétléses variációk száma). Következik, hogy ugyanennyi az n pontú cimkézett fák száma is. □

19.13. Definíció. Egy körmentes, de nem feltétlenül összefüggő gráf neve **erdő**. Azonnali, hogy egy erdő összefüggő komponensei fák (innen származik az elnevezés).

19.14. Feladat. Hány éle van egy n pontú, k komponensű erdőnek?

20. Feszítőfák, Kruskal-algoritmus

Ebben a szakaszban is egyszerű gráfokkal foglalkozunk.

20.1. Definíció. Egy összefüggő gráf **feszítőfájá**nak nevezzük a gráf olyan feszítő részgráfját, amely fa (lásd a 18.10. Definíciót).

Másképp: Egy összefüggő gráf feszítőfája a gráf olyan részgráfja, amely az adott gráf minden pontját tartalmazza (a gráf élei közül csak bizonyos éleket), és amely fa. A feszítőfa más elnevezése: faváz.

A 26. ábrán a bal oldalon egy 20 pontú összefüggő gráf látható (ugyanaz, mint a 20. ábrán), a jobb oldalon pedig ennek egy feszítő gráfja.

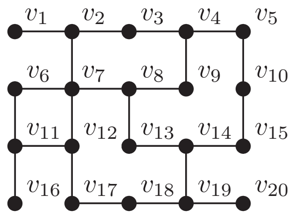
20.2. Tétel. *Minden összefüggő gráfnak létezik feszítőfája.*

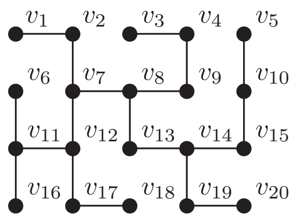
Bizonyítás. Legyen G egy tetszőleges összefüggő gráf. Ha G körmentes, akkor G egy fa és G lesz a feszítőfa. Ebben az esetben G az egyedüli feszítőfa.

Ha G nem összefüggő, akkor van benne egy C kör. A C kör egy e élét elhagyva $G \setminus \{e\}$ összefüggő marad, lásd 18.17. Feladat. Ha $G \setminus \{e\}$ körmentes, akkor egy fa, és ez lesz egy feszítőfa. Ellenkező esetben hagyjuk el egy kör valamelyik élét. Ezt ismételve véges sok lépés után a G egy feszítőfájához jutunk. □

Ha egy G összefüggő gráfban van kör, akkor G-ben több feszítőfa is van, és ezek között nem izomorfak is lehetnek.

20.3. Feladat. i) Adjuk meg a 26. ábrán levő bal oldali gráf más feszítőfáit.
ii) Adjuk meg a 16. ábrán levő első gráf néhány feszítőfáját. Keressünk nemizomorf feszítőfákat.





26. ábra

Nézzük a következő gyakorlati problémát: Adott n település, amelyek között vízvezeték hálózatot akarnak létesíteni úgy, hogy az minden települést ellásson az egyik A településen levő vízforrásból. Nem akarják feltétlenül A-t összekötni az összes többivel, a hálózaton keresztül azonban mindegyik számára biztosítani akarják a vízzel való ellátást. Az egyes települések közötti közvetlen vezetékek építési költségei ismertek (ezeket a távolság, a domborzati viszonyok és egyébb tényezők határozzák meg).

Kérdés: Mely településeket kell közvetlenül összekapcsolni úgy, hogy a teljes építési költség a lehető legkisebb legyen?

Gráfelméleti fogalmakat használva: Adott egy n pontú teljes gráf, amelynek minden éléhez hozzárendelünk egy pozitív valós számot, az építési költséget.

Feladat: Határozzuk meg ennek a gráfnak egy olyan részgráfját, amely
- összefüggő,
- minden pontot tartalmaz,
- éleihez rendelt költségek összege minimális.

Azonnali, hogy a keresett részgráf egy fa, mert összefüggő (feltétel) és körmentes, hiszen ha lenne benne kör, akkor annak legköltségesebb élét megszüntetve a részgráf összefüggő marad, az összköltség pedig csökken. Következik, hogy a keresett részgráf az adott n pontú teljes gráf egy feszítőfája, amelynek $n-1$ éle van.

Hogyan lehet egy ilyen feszítőfát megadni?

Tudjuk, hogy összesen n^{n-2} számú n pontú cimkézett fa létezik (Cayley-tétel). Ha sorra akarnánk venni a lehetőségeket, akkor pl. már $n = 10$-re $10^8 = 100\,000\,000$ lehetséges fát kellene megvizsgálnunk. Ennél egyszerűbb és hatékonyabb módszerre van szükség, és ilyen eljárás létezik.

A gyakorlatban bizonyos települések közötti közvetlen vezeték (kapcsolat) valamilyen meggondolás miatt ki van zárva. A teljes gráf helyett tehát adott egy összefüggő gráf, és ennek kell meghatározni egy minimális költségű feszítőfáját. Általában több ilyen fa létezik.

Bemutatjuk a Kruskal-algoritmust, amely alkalmas ennek a feladatnak a megoldására és amelynek lényege a következő: kiválasztunk a legkisebb költségű (legolcsóbb) élek közül egyet, majd ismételten a még ki nem választott, legolcsóbb élek közül kiválasztunk egy olyan élt, amely nem hoz létre kört a már kiválasztott élekkel.

Pontosabban, legyen G egy n pontú összefüggő gráf, amely éleinek halmaza $E(G)$ és legyen $c : E(G) \to (0, \infty)$ egy költségfüggvény a G élein. Határozzunk meg egy olyan F feszítőfát, amelyre a $c(F) = \sum_{e \in E(F)} c(e)$ teljes költség minimális.

Kruskal-algoritmus:

Rendezzük az éleket a költségek növekvő sorrendjében:

$E(G) = \{e_1, e_2, ..., e_m\}$, ahol $c(e_1) \le c(e_2) \le ... \le c(e_m)$.

Legyen $A = \emptyset \subseteq E(G)$, A lesz a kiválasztott élek halmaza.

Minden $i = 1, 2, ..., m$ esetén az e_i élt vagy kiválasztjuk, vagy sem az A halmazba a következőképpen:
- Ha az $A \cup \{e_i\}$ élhalmaz nem tartalmaz kört, akkor az e_i élt kiválasztjuk: $A := A \cup \{e_i\}$.
- Ha az $A \cup \{e_i\}$ élhalmaz tartalmaz kört, akkor az e_i élt nem választjuk ki, ekkor A nem változik.

Az algoritmus egy olyan F részgráfot eredményez, amelyet a G gráf összes pontja és az A-beli élek alkotnak ($E(F) = A$).

20.4. Tétel. *A Kruskal-algoritmus által eredményezett F részgráf egy minimális költségű feszítőfa.*

Bizonyítás. Az algoritmus során minden lépés után körmentes részgráfot kapunk, így a F output részgráf is körmentes.

Tegyük fel, hogy F nem összefüggő. Legyen F' az F egy összefüggő komponense. G összefüggő, ezért G-nek van olyan e_i éle, amelynek egyik végpontja F'-beli, a másik végpont pedig nem F-beli. Ez az e_i él nem került kiválasztásra, azaz $e_i \notin A$. Ugyanakkor ez az e_i él nem alkot kört F-ben, ezért az algoritmus során ezt ki kellett volna választani. Ez ellentmondást jelent.

Következik, hogy F egy fa, és mivel F pontjai között a G minden pontja szerepel (Indokoljuk meg!), ezért F a G egy feszítőfája.

Most igazoljuk, hogy F egy minimális költségű feszítőfa. Legyenek F élei $a_1, a_2, ..., a_{n-1}$ a kiválasztás sorrendjében, azaz $c(a_1) \leq c(a_2) \leq ... \leq c(a_{n-1})$. Legyen ugyanakkor T egy tetszőleges, F-től különböző feszítőfa, amely éleinek halmaza $E(T) = \{t_1, t_2, ..., t_{n-1}\}$.

Legyen a_i $(i \geq 1)$ az első olyan él az algoritmus során, amely nem éle T-nek (biztosan van ilyen él, mert $T \neq F$), tehát $a_1, a_2, ..., a_{i-1} \in E(T)$, de $a_i \notin E(T)$. Itt $i = 1$ is lehet, ebben az esetben már $a_1 \notin E(T)$.

Az a_i élt a T fához hozzávéve egy C kört kapunk (lásd 19.3. Tétel). Ez a C kör nem lehet teljes egészében F-ben, mert F egy fa, ezért C-nek létezik legalább egy olyan éle e éle, amely nem éle F-nek: $e \notin E(F)$, és éle T-nek: $e \in E(T)$. Legyen T_1 az a gráf, amelyet úgy kapunk T-ből, hogy T éleihez hozzávesszük a_i-t és elhagyjuk e-t, azaz $E(T_1) = E(T) \cup \{a_i\} \setminus \{e\}$. Akkor T_1 is egy feszítőfa (Igazoljuk ezt!).

Megmutatjuk, hogy itt $c(a_i) \leq c(e)$. Tegyük fel, hogy ez nem így van, azaz $c(e) < c(a_i)$. Ha itt $i = 1$, akkor $c(e) < c(a_1)$ ellentmond az algoritmus első lépésének. Ha $i = 2$, akkor $c(e) < c(a_2)$ ismét ellentmondás, mert e az a_1-gyel nem alkothat kört, ezért a második lépésben az e élt kellett volna választani. Legyen $i \geq 3$ és $c(e) < c(a_i)$. Az algoritmus az i-edik lépésben nem az „olcsóbb" e élt választotta, hanem a „drágább" a_i élt, aminek csak az a magyarázata lehet, hogy az e él kört alkot az F fa már kiválasztott $a_1, a_2, ..., a_{i-1}$ éleivel. De ezek az élek mind élei a T fának, ezért itt is ellentmondásra jutunk. Következik, hogy valóban $c(a_i) \leq c(e)$. Innen pedig azt kapjuk, hogy T_1 költsége nem több, mint T költsége: $c(T_1) \leq c(T)$.

Cseréljük ki a T fát a T_1 fára. A T_1 feszítőfának több közös éle van F-fel, mint T-nek, mert T_1-et úgy kaptuk, hogy T egy nem F-beli élét (az e élt) egy F-beli élre (az a_i élre) cseréltük. Ha T_1 különbözik F-től, akkor ugyanezt az eljárást ismételve olyan T_2, T_3,... feszítőfákat kapunk, amelyeknek egyre több közös élük van F-fel és ... $\leq c(T_3) \leq c(T_2) \leq c(T_1) \leq c(T)$.

Véges sok lépés (csere) után olyan T_k fát kapunk, amelyre $T_k = F$ és következik, hogy F költsége nem több, mint T költsége: $c(F) = c(T_k) \leq c(T)$. $\square$

Megjegyzések. 1. A fenti eljárást *mohó algoritmus*nak is nevezzük, mert az minden lépésben a még nem vizsgált élek közül a „legolcsóbbat" választja. Csak akkor nem használja a „legolcsóbbat", ha az már nem megengedett. Ez a „filozófia" sok esetben, így a fentiekben is (de nem mindig!) eredményre vezet.

2. Ha az eredeti G gráf élei mind különböző költségűek, azaz $c(e_1) < c(e_2) < ... < c(e_m)$, akkor a fentiek szerint következik, hogy a minimális költségű feszítőfa egyértelműen meghatározott.

3. Az *R. C. Prim által adott algoritmus* a következő: Legyen G egy összefüggő gráf. Tekintsük ennek egy a_1 tetszőleges pontját. Az a_1-ből kiinduló élek közül válasszunk egy minimális költségűt, így egy a_1, a_2 pontokból álló fát kapunk. Válasszunk egy olyan minimális költségű élt, amelynek egyik végpontja a_1 vagy a_2, a másik végpontja pedig egy ezektől különböző a_3 pont, így egy hárompontú fát kapunk. Ezt ismételve, ha már van egy $a_1, a_2, ..., a_k$ pontokból álló T fa, akkor válasszunk egy olyan minimális költségű élt, amelynek egyik végpontja T-beli, a másik végpontja T-n kívüli, és ezt az élt vegyük hozzá T-hez. Az eljárást addig folytatjuk, míg egy feszítőfát nem kapunk.

Igazolható, hogy az így kapott feszítőfa minimális költségű.

4. Egy más algoritmus: Legyen G egy összefüggő gráf. Rendezzük az éleket a költségek csökkenő sorrendjében: $E(G) = \{e_1, e_2, ..., e_k\}$, ahol $c(e_1) \geq c(e_2) \geq ... \geq c(e_k)$.

Legyen $B = E(G)$, B lesz a kiválasztott élek halmaza.

Minden $i = 1, 2, ..., k$ esetén az e_i élt vagy töröljük a B halmazból, vagy sem a következőképpen:

- Ha a B élhalmaz tartalmaz olyan kört, amelynek egyik éle e_i, akkor az e_i élt töröljük: $B := B \setminus \{e_i\}$.
- Ha a B élhalmaz nem tartalmaz olyan kört, amelynek egyik éle e_i, akkor B nem változik.

Igazolható, hogy ennek az algoritmusnak az outputja egy minimális költségű feszítőfa.