Preštevanje dreves

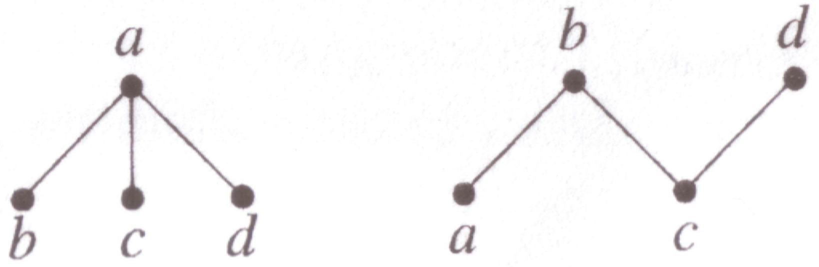
Vilma Šuštar

18. maj 2005

Obstaja $2^{\binom{n}{2}}$ enostavnih grafov množice točk $[n] = 1, \ldots, n$, če lahko vsak par točk tvori povezavo ali pa tudi ne. Koliko od teh grafov je dreves? V tem poglavju bomo rešili ta problem preštevanja in prešteli vpeta drevesa v poljubnem grafu.

SEZNAM DREVES

Z eno ali dvema točkama lahko tvorimo le eno drevo. Pri treh točkah imamo še vedno samo en razred homeomorfnih dreves, ampak **matrika sosednosti** je določena s tem, katera točka je center. Zato množica točk $n = 3$ tvori tri drevesa. Pri množici točk $n = 4$ imamo 4 točke in 12 povezav, kar predstavlja 16 dreves. Podobno bi za množico točk $n = 5$ ugotovili, da lahko tvorimo 125 dreves.

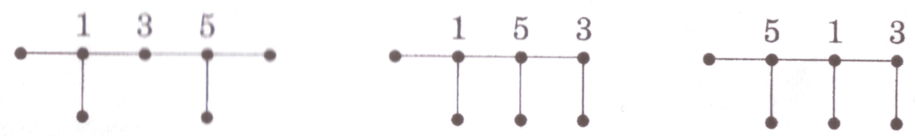


Sedaj lahko vidimo vzorec: pri n točkah imamo n^{n-2} dreves. To je **Cayleyeva formula**. Dokazali so jo Prüfer, Kirchhoff, Pólya, Reny in drugi. Mi bomo naredili dokaz, ki temelji na bijekciji med množico dreves z n točkami in množico z znano velikostjo.

Podana je množica S z n števili. Potem lahko na natanko n^{n-2} načinov tvorimo seznam dolžine $n-2$ s podatki iz množice S. Množico seznamov označimo s S^{n-2}. S^{n-2} uporabljamo za kodiranje dreves z množico točk S. Seznam, ki ga na tak način priredimo trevesu je njegova **Prüfova koda**.

Algoritem 1. *(Prüfova koda)*
Vhod: *Drevo T z množico točk $S \subseteq \boldsymbol{N}$.*
Izhod: *koda $f(T) = (a_1, a_2, \ldots, a_{n-2})$*
Ponavljanje: *na i-tem koraku, izbriši najmanjši preostali list, in naj bo a_i njegov sosed.*

PRIMER 1. *Po $n-2$ ponovitvah ostane samo ena od začetnih $n-1$ povezav. Dobili pa smo seznam $f(T)$ dolžine $n-2$ s podatki iz S. V drevesu spodaj je najmanjši list 2, zbrišemo ga in zapišemo v seznam 7. Potem zbrišemo 3 in 5 in vsakič zabeležimo 4. Najmanjši list, ki nam je ostal v drevesu na 5 točkah je 4 ...*
Koda, ki jo dobimo je (744171) in točki, ki na koncu ostaneta sta 1 in 8. Po prvem koraku je ostanek Prüferjeve kode Prüferjeva koda poddrevesa T' z množico točk $V(T) - \{2\}$.



Če poznamo množico točk S, lahko iz kode a nazaj dobimo drevo. Ideja je v tem, da najdeš vse povezave. Začnemo z množico izoliranih točk S. Na vsakem koraku naredimo eno povezavo in označimo eno točko. Ko razmišljamo o a_i, nam ostane $n-i+1$ neoznačenih točk in $n-i-1$ znakov iz a (vključno z a_i). Tako se vsaj dve od neoznačenih točk ne pojavljata med ostalimi znaki v a. Naj bo x najmanjši od teh znakov. Dodaj povezavo xa_i in označi x. Po $n-2$ korakih, nam ostaneta dve neoznačeni točki; povežemo ju in s tem tvorimo zadnjo povezavo.

Ogeljmo si zgornji primer. Najmanjši element množice S, ki ni v kodi a je 2. Tako bo prva povezava, ki jo bomo dodali povezovala 2 in 7, in označimo 2. Sedaj je najmanjši element, ki manjka v kodi 3. Zato povežemo 3 z 4, ki je a_2. Ko nadaljujemo, obnavljamo povezave po vrstnem redu, kot so bile izbrisane, ko smo iskali a iz T.

V tem procesu ima vsaka komponenta grafa, ki smo ga gradimo eno neoznačeno točko. Če bi dodali povezavo med tema dvema neoznačenima točkama, bi med seboj povezali dve komponenti grafa. Po tem, ko označimo *eno točko nove povezave, ima zopet vsaka komponenta eno neoznačeno točko. Po $n-2$ korakih, imamo dve neoznačeni točki in zato dve komponenti. S tem, ko dodamo zadnjo povezavo med tema dvema točkama, tvorimo povezan graf. Zgradili smo povezan graf z n točkami in $n-1$ povezavami. Po eni prejšnjih trditev, je to res drevo, nismo pa še dokazali, da je njegova Prüfova koda enaka a.*

Izrek 1. *(Cayley's formula [1889]). Za množico $S \subseteq \boldsymbol{N}$ z močjo n, obstaja n^{n-2} dreves s točkami iz S.*

Dokaz. (Prüfer [1918])
Za $n = 1$ to drži, zato predpostavimo $n \geq 2$. Dokažimo, da zgornji algoritem določa bijekcijo f iz množice dreves z množico točk S v množico S^{n-2}, ki je množica seznamov dolžine $n-2$ iz množice S. Za vsak $a = (a_1, a_2, \ldots, a_{n-2}) \in S^{n-2}$ moramo pokazati, da natanko eno drevo T z množico točk S zadošča $f(T) = a$. Dokažimo to z indukcijo po n. Baza indukcije: $n = 2$. Imamo eno drevo z dvema točkama. Prüferjevava koda je seznam dolžine 0 in je edini tak seznam.

Indukcijski korak: $n > 2$. Računanje $f(T)$ zniža stopnjo vsake točke na 1 in jo potem po možnosti izbriše. Zato se vsaka točka, ki ni list pojavi v $f(T)$. V $f(T)$ pa ni nobenega lista, ker bi to pomenilo, da je list sosed lista, kar pa je možno le v drevesu z eno samo točko, mi pa smo predpostavili, da imamo več kot 2 točki. Zato so listi T elementi S, niso pa nujno elementi $f(T)$. Če $f(T) = a$, potem je prvi list, ki ga izbrišemo, najmanjši element iz S, ki ni v a (imenumo ga x), in sosed od x je a_1.

Dan imamo $a \in S^{n-2}$ in iščemo vse rešitve $f(T) = a$. Pokazali smo že, da ima vsako tako drevo točko x kot najmanjši list in povezavo xa^1. Ko izbrišemo x nam ostane drevo z množico točk $S' = S - \{x\}$. Njegova Prüfova koda je $a' = (a_2, \ldots, a_{n-2})$.

Po indukcijski predpostavki obstaja natanko eno drevo T' na množici točk S' in s Prüferjevo kodo a'. Ker vsako drevo s Prüferjevo kodo a dobimo tako, da takšnemu drevesu dodamo povezavo xa_1, obstaja največ ena rešitev za $f(T) = a$. Še več, če T' dodamo povezavo xa_1, dobimo drevo z množico točk S in Prüferjevo kodo a, torej obstaja vsaj ena rešitev. In zato obstaja natanko ena. □

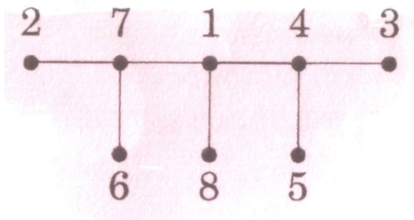
Posledica 2. *Naj bodo $d_1, d_2, \ldots, d_n$ pozitivna cela števila, katerih vsota je $2n-2$. Potem obstaja natanko $\frac{(n-2)!}{\Pi(d_i-1)!}$ dreves z n točkami takšnih, da ima točka i stopnjo d_i za vsak i.*

Dokaz. Med konstruiranjem Prüferjeve kode drevesa T, zapišemo x vsakič, ko izbrišemo soseda od x, dokler ne izbrišemo x samega ali ga pustimo med zadnjima dvema točkama. Tako se vsaka točka x pojavi $d_T - 1$ krat v Prüferjevi kodi.

Zato štejemo drevesa s stopnjami teh točk tako, da štejemo sezname dolžine $n-2$, ki ima za vsak i $d_i - 1$ kopij i. Če kopijam vsakega i dodamo indekse, da jih razlikujemo, potem permutiramo $n-2$ različnih predmetov in imamo $(n-2)!$ seznamov. Dokler nismo razlikovali kopij i, smo šteli vse željene razvrstitve $\Pi(d_i - 1)!$ krat. $\square$

PRIMER 2. *Drevesa s stalno stopnjo*

Premislimo o drevesih s točkami $\{1,2,3,4,5,6,7\}$*, ki so po vrsti stopnje* $\{3,1,2,1,3,1,1\}$. *Izračunajmo* $\frac{(n-2)!}{\Pi(d_i-1)!} = 30$*; drevesa so predstaljvena spodaj. Samo točke* $\{1,3,5\}$ *niso listi. Ko zbrišemo liste dobimo poddrevo na točkah* $\{1,3,5\}$. *Imamo tri taka poddrevesa, določena s tem, katera točka je na sredini.*



Da dopolnimo vsako drevo, dodamo primerno število sosednjih listov vsaki točki, ki ni list, da dobi željeno stopnjo. Šest možnosti je, da dopolnimo prvo drevo (iz preostalih štirih točk izberemo dve, ki bosta sosednji točki 1*) in dvanajst možnosti, da dopolnimo vsakega od ostalih dveh grafov (izberemo sosednjo točko točke* 3 *iz preostalih štirih točk in potem soseda centralne točke od preostalih treh).*