==

Stupně uzlů

==

Není-li výslovně uvedeno jinak, uvažujeme vždy konečný neorientovaný prostý graf bez smyček.

Musí být počet uzlů lichého stupně v grafu sudý nebo lichý? Proč?
[Je sudý. Každá hrana má dva konce, čímž přispívá hodnotou 2 do celkového součtu stupňů, jenž je tím pádem sudý. Kdyby počet uzlů lichého stupně v grafu byl lichý, byl by lichý i celkový součet stupňů, což nejde.]

Platí předchozí konstatování o součtu stupňů také obráceně, tj. může každé sudé číslo představovat součet stupňů všech uzlů nějakého grafu G? Nebo některá sudá čísla nemohou být uvedeným součtem? Která?
[Na cestě délky n je součet stupňů roven $2n$.]

Může existovat graf, jenž má méně hran než uzlů a neobsahuje vrchol stupně 1?
[K trojúhelníku K_3 přidáme izolovaný uzel.]

Když se dva účastníci konference setkají poprvé, vymění si navštívenky. Je počet navštívenek, které takto změnily majitele během konference, sudý nebo lichý? Je sudý nebo lichý počet účastníků, kteří rozdali lichý počet navštívenek?
[Sudý, sudý. Ve druhém případě platí, že počet uzlů lichého stupně v každém grafu je sudý.]

Turnaj hraje 8 družstev systémem každý s každým. Bylo odehráno celkem 9 zápasů. Je možné, že každé družstvo sehrálo zatím nejvýše 2 zápasy? Je možné že po odehrání celkem 17 zápasů sehrálo každé družstvo nejvýše 4 zápasy?
[Každé družstvo nejvýše 2 zápasy $\Rightarrow$ každé družstvo uzel grafu se stupněm max 2 $\Rightarrow$ počet hran v grafu nejvýše 8, čili 9 zápasů není možných. Podobně pro 17 zápasů to není možné, $8{\cdot}4 / 2 < 17$.]

Mohou být všechny stupně uzlů v obyčejném grafu navzájem různá čísla?
[Ne. V souvislém grafu jsou možné stupně jen 1,2 ... $n-1$, což je méně než n možností. Samostatně doplňte argument pro nesouvislé grafy.]

Daný graf s n uzly má právě dva vrcholy stejného stupně. Může to být stupeň 0 nebo $n-1$?
[Ne, ani 0 ani $n-1$.]

Dokažte pomocí pokud možno co nejjednodušší konstrukce, že pro každé $n>2$ existuje graf s n uzly, který obsahuje rovnoběžné hrany (nikoli smyčky) a jehož všechny uzly mají různý stupeň.
[Např. Vezmeme libovolný souvislý prostý graf, hrany očíslujeme 1..m, a na místě hrany j uděláme 2^j paralelních hran.]

Najděte graf se 7 uzly, v němž právě dva uzly mají stejný stupeň. Najděte ještě jiný takový graf. Jsou navzájem svými doplňky?
[Uzly: A, .. G, hrany: BG, CF, CG, DE, DF, DG, EF, EG, FG. Druhý graf je doplňkem.]

Otázka, zda ve společnosti mohou být zároveň dvě trojice, kde v jedné se všichni znají a v druhé všichni neznají, je triviální. Na pozitivní odpověď stačí pět osob. Nakreslete příslušný graf.
[Osoby A,B,C se navzájem znají, D se zná jen s A, E se zná jen s B. Osoby D,E,C se navzájem neznají.]

Může mít souvislý graf méně hran než uzlů?
[Ano, kterýkoli strom.]

Může mít nesouvislý graf více hran než uzlů?
[Ano, k úplnému grafu s alespoň 4 uzly připojme izolovaný uzel.]

Jaký minimální počet hran je nutno odebrat z K_n, aby vznikl nesouvislý graf?
[Všechny hrany incidentní s jedním uzlem, jichž je $n-1$.]

Jaký maximální počet hran lze odebrat z Kn, aby zůstala zachována souvislost grafu?
[Rozdíl počtu hran Kn a počtu hran ve stromu s n uzly, tj. $n(n-1)/2 - (n-1) = (n-1)(n-2)/2$.]

Jaký minimální počet hran je nutno odebrat z K_n, aby vznikl graf s k komponentami?
[Postupným oddělováním jednotlivých uzlů, $(n-1) + (n-2) + ... + (n-k) = k(2n-k-1)/2$.]

==
Stromy
==

Jsou dány tři kusy papíru. Některý z nich rozstříhneme na tři kusy, některý ze všech kusů rozstříhneme opět na tři kusy, atd. Kolik kusů celkem vznikne, když bylo rozstřiženo celkem k kusů papíru (nezávisle na velikosti)? Řešte úlohu rovněž obecně pro m počátečních kusů a stříhání na n dílů. Interpretujte úlohu grafově.
[Rozstřižení papíru na n kusů je vnitřní uzel stupně n kořenového stromu, výsledné kousky představují jeho listy. Každým rozstříhnutím přibyde $n-1$ kusů, ve výsledku máme $m + k\cdot(n-1)$ kusů.]

Na stole vidíme ležet k uzavřených krabic. Kdykoli otevřeme jakoukoli krabici, bude buď prázdná nebo bude obsahovat opět k (nejspíše menších) uzavřených krabic. Kromě toho víme, že celkový počet neprázdných krabic je m. Když otevřeme všechny plné krabice, kolik uvidíme uzavřených krabic?
[Zadání říká, že počet vnitřních uzlů kromě kořene pravidelného kořenového stromu stupně k je celkem m. Počet listů, což jsou neotevřené prázdné krabice, je $k + m\cdot(k-1) = k+m\cdot k-m$.]

Turnaje ve stolním tenisu, který se hraje vylučovacím způsobem, se zúčastnilo celkem 19 hráčů. Šest vylosovaných hráčů muselo sehrát o jedno utkání více. Nakreslete graf reprezentující průběh turnaje a určete počet jeho vnitřních uzlů tj. počet utkání, která byla během turnaje sehrána.
Kolik utkání by bylo nutno sehrát pro 147 hráček a hráčů?
[Utkání: 3+15 = 18, 19 + 127 = 146.]

Ve stromu s n uzly určíme každé hraně náhodnou orientaci. Jaká je pravděpodobnost, že vznikne kořenový strom?
[$n/2^{n-1}$. Když zvolíme kořen stromu, je orientace hran jednoznačně dána. Možných kořenů je celkem n, to je tedy počet všech příznivých případů. Počet všech možných případů (orientací hran) je $2^{\text{počet hran}} = 2^{n-1}$.]

==
Algoritmy a implementace
==

Úplný binární strom se sedmi uzly je vyvážený a uzly jsou očíslovány po patrech zleva doprava 1...7 (jako v binární haldě). Napište matici sousednosti tohoto grafu. Předpokládejme, že náš strom obsahuje $2^n - 1$ uzlů ($n > 0$), tj. má hloubku $n-1$ (kořen má hloubku 0). Popište algoritmus, jímž matici sousednosti vygenerujete přímo, bez skutečného procházení tímto stromem.
[Stačí použít indexaci haldy v poli, prvek na pozici i má potomky na pozicích $2i-1$, $2i$, tedy jedničky v matici sousednosti budou právě na pozicích $(i, 2i-1)$, $(i, 2i)$.]

Druhá mocnina grafu G je graf G^2, jehož množina uzlů se shoduje s množinou uzlů grafu G a jehož množina hran je určena takto: G2 obsahuje hranu (u,v) jen a jen tehdy, pokud G obsahuje zároveň hrany (u,w) a (w,v), kde w je libovolný uzel grafu G. Jinými slovy, G^2 vznikne z G tak, že do G přidáme hrany mezi všemi uzly spojenými cestou délky 2 a odstraníme původní hrany. Popište, jak vytvoříte G^2, když jsou grafy zadány (a) spojovou reprezentací (b) maticí sousednosti. Která varianta bude rychlejší?

Modifikujte algoritmus (nebo váš funkční program) DFS tak, aby vypsal každou hranu grafu (každou právě jednou).

Řešte předchozí úlohu pro BFS.

Napište funkční DFS kód rekurzivně i nerekurzivně.