CVIČENÍ 7

Téma: Vlastnosti stromů, algoritmy na stromech.
Cíle: Zvládnout vzájemnou provázanost tří hlavních vlastností stromů (graf, který je: souvislý, bez kružnic a má o jednu hranu méně než uzlů), dokázat vhodně upravit základní grafové algoritmy, pokud se mají týkat stromů.

Stromy	
1.	Kolik může mít maximálně, resp. minimálně listů neorientovaný strom s celkem n (≥ 2) uzly?
2.	Nechť **T** je neorientovaný strom, v němž je maximální stupeň uzlu roven $\boldsymbol{\delta_{max}}$($\geq 2$). Ukažte, že strom **T** má nejméně $\boldsymbol{\delta_{max}}$ listů (tj. uzlů stupně 1).
3.	Nechť **T** je neorientovaný strom mající přesně **k** listů, jehož všechny vnitřní uzly mají stejný stupeň **d**(≥ 3). Určete počet vnitřních uzlů stromu **T** v závislosti na **k** a **d**.
4.	Navrhněte algoritmus, který pro zadaný obyčejný neorientovaný graf G = $\langle$H,U$\rangle$ zjistí v čase O(|U|), zda graf G obsahuje nějakou kružnici.
5.	Neorientovaný strom **S** je zadaný maticí sousednosti **V**. Sestavte algoritmus výpočtu jeho poloměru, průměru a středů z této matice. Určete složitost tohoto algoritmu. Změňte reprezentaci stromu na pole ukazatelů na seznamy sousedů a navrhněte skutečně efektivní algoritmus (podobný jako při testování acykličnosti, ale pozor na to, že listy se vždy musí odebrat "naráz").
6.	Nechť $\mathbf{T_1}$ a $\mathbf{T_2}$ jsou dvě hranově disjunktní kostry (souvislého) neorientovaného grafu **G** s **n** uzly. Je možné určit minimální počet kružnic, které vzniknou sjednocením $\mathbf{T_1} \cup \mathbf{T_2}$?
7.	Nechť $\mathbf{T_1}$, resp. $\mathbf{T_2}$ je strom prohledání souvislého neorientovaného grafu **G** do šířky, resp. do hloubky. Je možné z nějakých vlastností stromů $\mathbf{T_1}$ a $\mathbf{T_2}$ usuzovat něco o poloměru **r(G)** a průměru **T(G)** grafu **G**?
8.	V neorientovaném stromu o n uzlech provedeme náhodně orientaci hran. Jaká je pravděpodobnost, že vznikne kořenový strom?
9.	V úplném neorientovaném grafu provedeme náhodně orientaci hran. • Dokažte, že vzniklý graf má kořenovou kostru. • Pokud výchozí graf není úplný, lze naopak nalézt taková orientace, že výsledný orientovaný graf nemá kořenovou kostru (určete jak).
10.	Potvrďte nebo vyvraťte následující tvrzení: Při procházení binárního stromu libovolným ze způsobů preorder, inorder, postorder se listy stromu projdou vždy ve stejném pořadí.
11.	Nalezněte obecný tvar binárních stromů, jejichž uzly se projdou ve stejném pořadí při průchodech • preorder a inorder • preorder a postorder • inorder a postorder
12.	Nechť **T** je kořenový strom s uzly očíslovanými **1, 2, ..., n**, který je reprezentován pouze pomocí odkazů **pred[u]** každého uzlu na svého předchůdce ve stromu, pro kořen **r** je **pred[r]=0**. Navrhněte algoritmus systematického průchodu stromem **T** s časovou složitostí **O(|U|)**, jehož cílem je • určit hloubku (tj. vzdálenost od kořene) všech uzlů • určit celkovou hloubku stromu **T** • určit maximální šířku stromu **T**