

ЗАДАЧИ ЗА ЗАДЪЛЖИТЕЛНА САМОПОДГОТОВКА ПО Структури от данни и програмиране

email: kalin@fmi.uni-sofia.bg

16 януари 2018 г.

1. Даден е граф $G :< V = \{0..n-1\}, E \subseteq V \times V \times \{a..z\} >$ с етикети на ребрата $\{a..z\}$, представен с матрица $std :: vector < char > G[n][n]$. Каваме, че думата $w = w_1, \dots, w_k$ може “да се прочете” в графа, ако в него има път между произволни два върха, състоящ се от последователни ребра с етикети w_1, \dots, w_k . Да се дефинира функция, която намира по колко различни начина може да се прочете дадена дума в даден граф (т.е. колко различни пътя в графа отговарят на това условие).
2. Даден е граф $G :< V = \{0..n-1\}, E \subseteq V \times V >$, представен с матрица $bool\ G[n][n]$. Нека $v \in V$ е връх в графа, а $k \in \mathbb{N}$. Да се построи и отпечата в `doty` формат дърво с корен v и височина най-много k , за което е изпълнено:
 - Всяко ниво $0 \leq l < k$ съдържа всички върхове u от графа, до които има път от v с дължина $l + 1$ (в брой върхове).
 - За всяко $0 < l < k$, елементът u на ниво $l - 1$ е родител в дървото на елемента w на ниво l тогава и само тогава, когато $(u, w) \in E$.
3. Даден е ориентиран ациклически граф $G :< V = \{0..n-1\}, E \subseteq V \times V >$, представен с матрица $bool\ G[n][n]$. Да се построи топологично сортиран вектор (v_1, \dots, v_n) от всички върхове на графа.
4. Даден е лабиринт, представен с матрица $bool\ L[n][n]$. Елементите $L[i][j] == true$ считаме за проходими, а елементите $L[i][j] == false$ - за непроходими.

Да се построи матрицата на съседство $bool\ G[n \times n][n \times n]$ на графа $G :< V, E >$, за който множеството V съдържа представители на всички проходими елементи на L , а $(u, v) \in E$ тогава и само тогава, когато u и v са проходими и са съседни в L .

Да се отпечата графа в dotty формат и да се намери броя на свързаните му компоненти.