Tema 1, 1 noiembrie 2019

Termen de predare: 8 noiembrie 2019, 12:00-14:00 în C210

- **Tema poate fi rezolvată în echipe de câte doi studenţi.**
- **Pentru soluţii redactate în LaTeX se oferă un bonus de 1 punct.**
- **Soluţiile identice sau copiate conduc la anularea punctajelor.**
- **Soluţiile trebuie să conţină numele celui/celor care au redactat-o şi vor fi transmise pe hârtie, nu prin e-mail.**
- **Nu rescrieţi enunţurile! Nu este nevoie de mai mult de $1-2$ pagini pentru fiecare dintre probleme.**
- **ORICE SOLUŢIE COPIATĂ A UNEIA DINTRE PROBLEMELE DE MAI JOS VA FI PENALIZATĂ CU 2 PUNCTE.**

1. Se consideră reţeaua stradală a unui oraş care este conexă - din orice intersecţie se poate ajunge în orice altă intersecţie (pe toate străzile se poate circula în ambele sensuri).

(a) Primăria oraşului doreşte să transforme fiecare stradă într-un sens unic aşa încât reţeaua stradală să rămână conexă - din orice intersecţie să se poată ajunge în orice altă intersecţie. Arătaţi că acest lucru e posibil dacă şi numai dacă prin blocarea oricărei străzi reţeaua stradală nu se deconectează.

(b) Descrieţi un algoritm care să întreprindă o astfel de transformare dacă este posibilă. Care este complexitatea sa timp?

(2 + 2 = 4 puncte)

2. Fie $G = (V, E)$ un graf conex şi $u, v \in V$ două noduri distincte ale lui G. O submulţime de noduri X se numeşte uv**-separatoare minimală** dacă u şi v se află în componente conexe diferite ale lui $G - X$, dar pentru orice $X' \subsetneq X$, u şi v sunt în aceeaşi componentă a lui $G - X'$.

(a) Dovediţi că $X \subseteq V$ este muţime uv-separatoare minimală dacă şi numai dacă u şi v se află în componente diferite ale lui $G - X$, iar orice nod din X are vecini în ambele aceste componente.

(b) Dacă X_1 şi X_2 sunt două mulţimi uv-separatoare minimale din G astfel încât X_1 intersectează cel puţin două componente din $G - X_2$, atunci X_1 intersectează componentele lui $G - X_2$ care conţin pe u şi v.

(2 + 2 = 4 puncte)

3. Fie $G = (V, E)$ un graf cu n noduri şi m muchii. Considerăm următorul algoritm:

$G' \leftarrow G$;
while ($\exists u \in V(G')$ a. î. $d_{G'}(u) < m/n$) **do**
  $G' \leftarrow G' - u$;
return G';

(a) Determinaţi complexitatea timp a unei implementări eficiente a acestui algoritm.

(b) Arătaţi că graful returnat, G', este nenul (are şi noduri, dar şi muchii).

(c) Arătaţi că orice graf conţine un drum de lungime cel puţin m/n.

(2 + 1 + 1 = 4 puncte)

4. Fie $G = (V, E)$ un digraf, $a : E \to \mathbb{R}_+$ o funcţie de cost pe arcele sale şi $x_0 \in V$ un nod din care toate celelate noduri ale lui G sunt accesibile. Un **SP-arbore** pentru tripleta (G, a, x_0) este un arbore cu rădăcină al lui G, $T = (V, E')$, aşa încât costul (cu aceeaşi funcţie a) drumului de la x_0 la u în T este costul minim al unui drum de la x_0 la u în G, pentru orice $u \in V$.

(a) Arătaţi că un astfel de SP-arbore există întotdeauna.

(b) Descrieţi un algoritm care să determine un SP-arbore.

(2 + 1 = 3 puncte)