

TEMA 2 - fiecare problema din teme are 0,15p

1. Se dă un graf orientat ponderat G fără circuite (DAG, ponderile sunt numere reale).
 - a) Modificați pseudocodul algoritmului de determinare a unui drum minim de la s la t în DAG pentru a determina numărul de drumuri minime între două vârfuri date s și t din G .
 - b) Descrieți un algoritm de determinare a numărului de drumuri dintre două vârfuri date s și t din G și scrieți pseudocodul lui. Justificați corectitudinea și complexitatea algoritmului propus.
2. Fie G un graf conex ponderat și T_1 și T_2 doi apcm în G . Fie L_1 lista sortată a costurilor muchiilor din T_1 (!doar a costurilor) și L_2 lista sortată a costurilor muchiilor din T_2 . Este adevărat că $L_1=L_2$? Justificați.
3. a) Dacă oprim algoritmul lui Dijkstra când mai este un singur vârf în Q (neselectat), rezultatul obținut este corect? Justificați.
b) Fie G un graf orientat ponderat și s un vârf în G . Dacă toate arcele au cost pozitiv cu excepția arcelor care ies din s care pot avea și cost negativ (dar fără a forma circuite negative), rezultatul furnizat de algoritmul lui Dijkstra mai rămâne corect? Justificați.
4. a) Fie $G = (V, E, w)$ conex cu ponderi distincte. Fie T_m unicul apcm al lui G . Fie T_s un arbore "second best" al lui G , adică un arbore cu cost minim dintre arborii parțiali ai lui G diferiți de T_m . Arătați că există două muchii $uv \in E(T_m)$ și $xy \notin E(T_m)$ astfel încât $T_s = T_m - uv + xy$
b) Fie G un graf conex ponderat cu ponderi distincte și T_m un apcm în G (ambele date prin lista de muchii cu costul asociat). Descrieți pe scurt un algoritm bazat pe rezultatul de la a) pentru a determina un arbore "second best" al lui G .
5. Fie G un graf conex ponderat și T_m un apcm în G (ambele date prin lista de muchii cu costul asociat). Descrieți un algoritm eficient care, date o muchie e din G și un cost nou c pentru această muchie, determină un apcm în graful obținut din G atribuind costului muchiei e valoarea c . Care este complexitatea acestui algoritm? Justificați corectitudinea algoritmului propus.