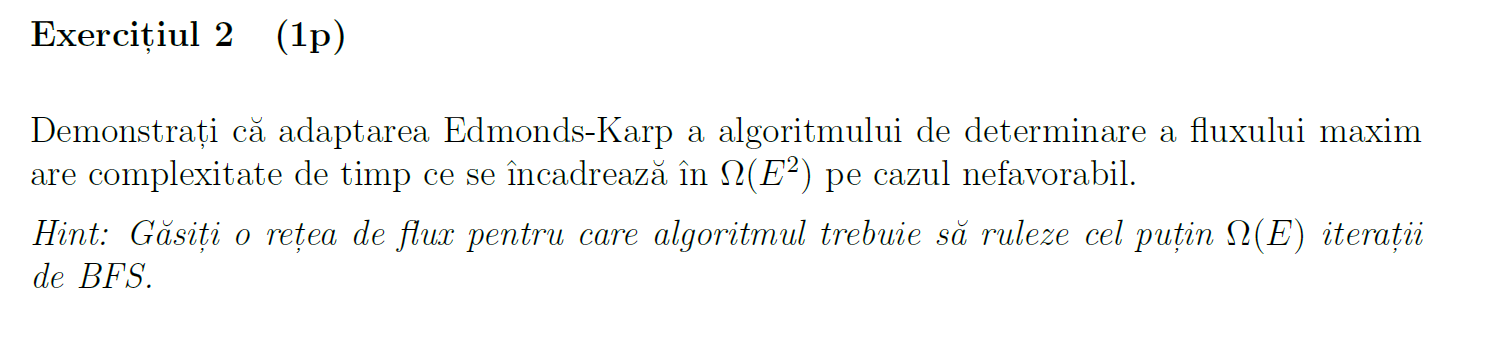
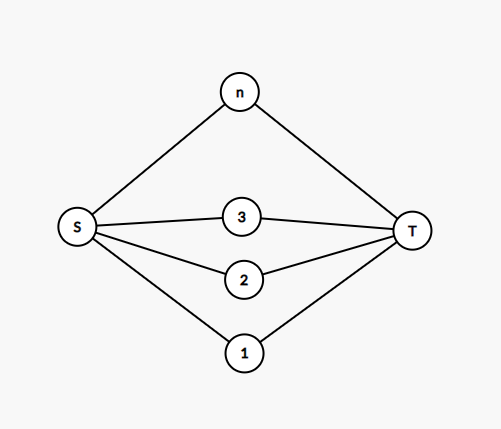
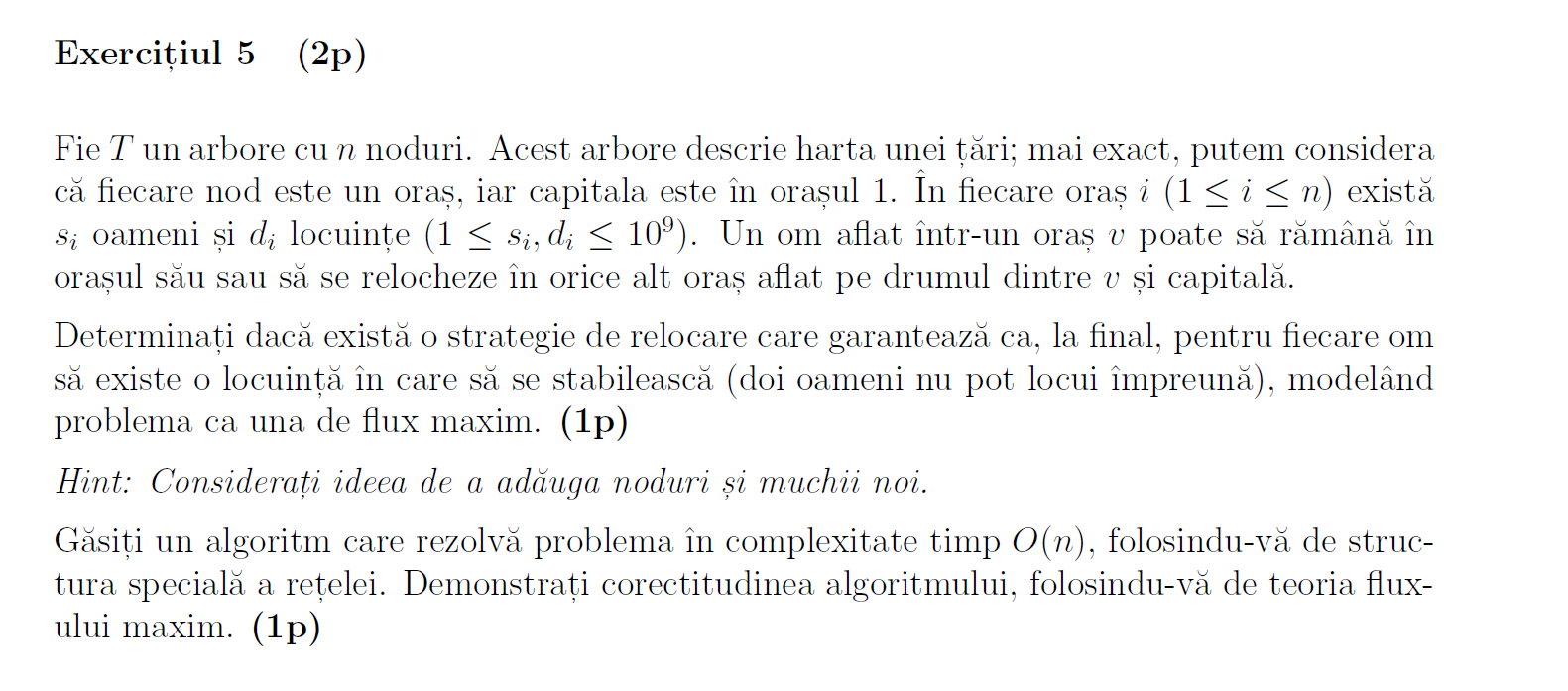
**Tema seminar 7**

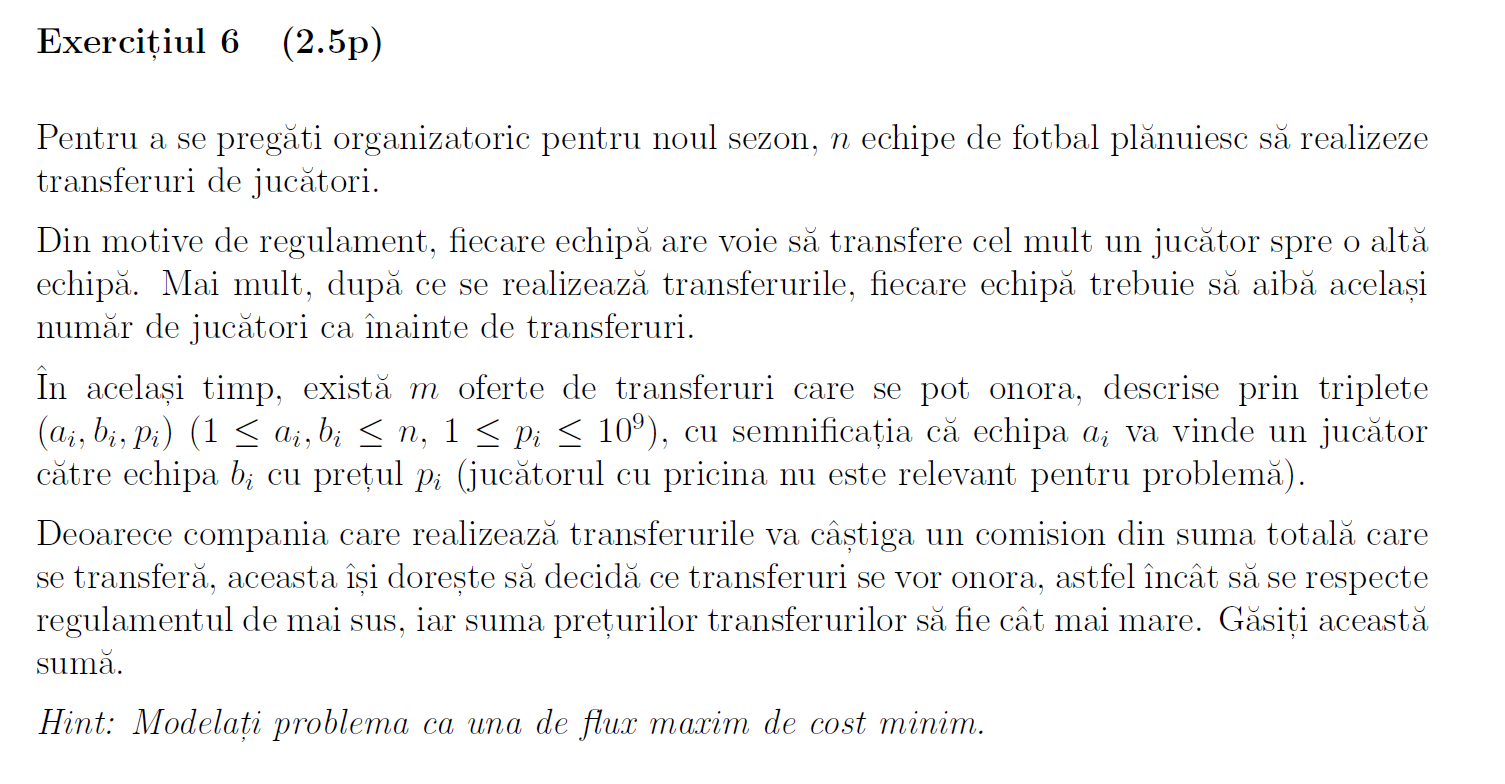


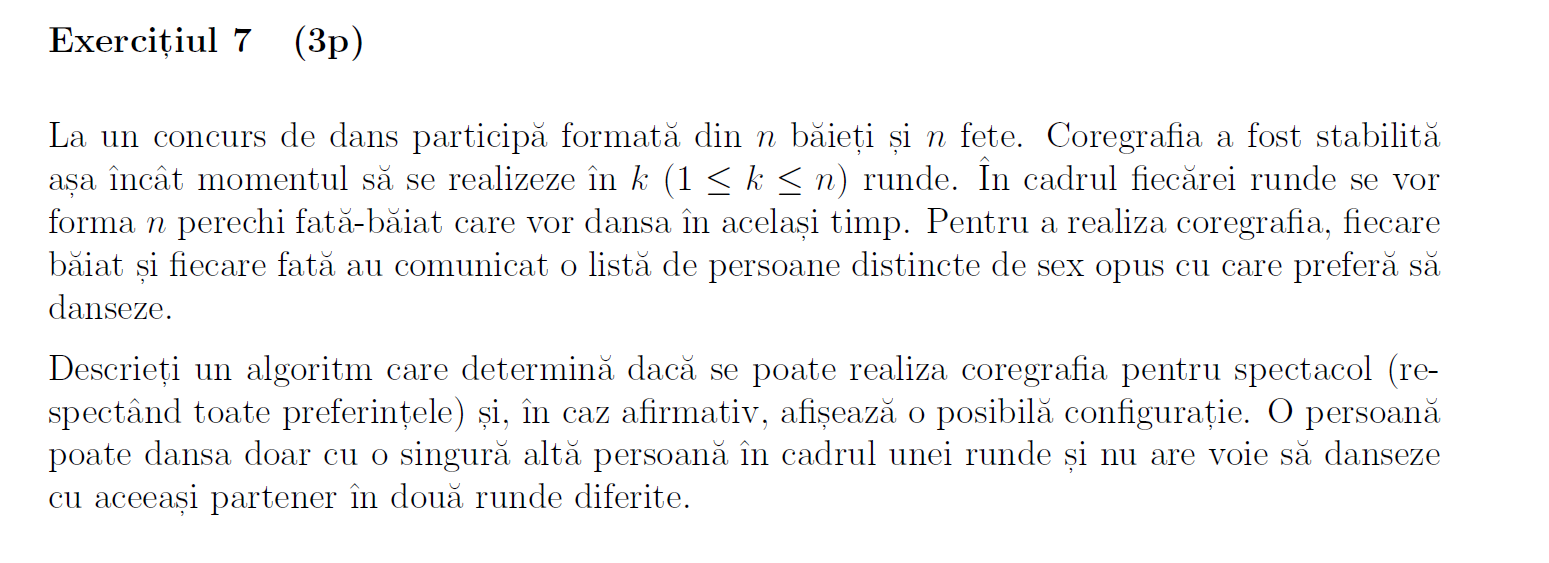


Algoritmul Edmonds-Karp foloseste parcurgeri BFS si ii putem asocia complexitatea Ω(E) pentru ca trebuie sa parcurga fiecare muchie. Un exemplu de retea de flux pentru care algoritmul trebuie sa ruleze cel putin Ω(E) iteratii de BFS este cel din imagine. Mai exact, parcurgerea grafului necesita Ω(E/2) iteratii de BFS pentru ca va face o augumentare pentru fiecare nod si numarul de noduri e dublul numarului de muchii. Pentru ca Ω(E/2)≈Ω(E) rezulta o complexitate totala de Ω(E2).

Pentru a rezolva problema folosind teoria fluxului maxim e necesar sa adaugam un nod sursa S si unul destinatie T. Legam apoi nodul S de toate orasele si punem capacitatea muchiei de la S la nodul i egala cu si. Apoi legam toate orasele de nodul D si punem capacitatea muchiei de la nodul i la T egala cu di. Intre oricare alte doua orase se vor pune muchii cu capacitate infinite pentru ca pot circula oricate persoane.

O persoana care vrea sa isi gaseasca casa e echivalentul a unei unitati de flux. Daca aceasta s-a stability e echivalentul a trimiterii unei unitati de flux in T. Daca fluxul maxim e egal cu numarul de oameni inseamnca ca exista o strategie de relocare.

Pentru a modela problema ca una de flux maxim de cost minim e necesar sa dublez numarul de noduri din graf si sa adug un nod sursa S si un nod destinatie T. O sa obtin un graf bipartite cu n noduri in stanga si n noduri in dreapta. Conectez S la toate nodurile din stanga si toate nodurile din dreapta la T cu muchii de cost 0 si capacitate 1. Intre echipele in care se poate face un transfer vom duce o muchie de cost -pretul transferului si capacitate 1 si ducem muchii de cost 0 daca nu se pot face transferuri. La final vom obtine -1 \* fluxul maxim de cost minim.



Vom avea un graf biparit intre cei n baieti si n fete si vom pastra muchiile care sunt atat de la baiat spre fata cat si de la fata spre baiat. Apoi asociez capacitatea 1 muchiilor si adaug un nod sursa S si un nod destinatie T. De S leg toti baietii si de T leg toate fetele iar capacitatea muchiilor va fi numarul de runde pe care trebuie sa il dansese persoana respectiva. Ca sa se poata realiza coregrafie trebuie ca la final sa obtinem un flux de n \* k.

Putem rula algoritmul de cuplaj de k ori si sa afisez perechiile rezultate urmand sa sterg muchiile folosite pentru ca ele nu se pot folosi decat intr-o singura coregrafie.

* Gasiti numarul minim de muchii pe care trebuie sa-l adaugati intr-un graf **orientat** astfel incat sa formati un graf eulerian

Se disting doua cazuri:

* Exista o singura componenta conexa in graf
* Sunt mai multe componente conexe in graf

In primul caz, daca toate nodurile din graf au grad par atunci el este deja eulerian si nu trebuie adugate muchii. Daca exista noduri cu grad impar atunci stim ca exista un numar par de noduri cu grad impar. Numarul minim de muchii este numarul de noduri cu grad impar impartit la 2.

In al doilea caz, putem marca componentele ca fiind pare si impare cu intelesul ca este impara daca contine macar un nod cu grad impar in ele. La componentele pare selectam un nod random din toate si le asezam intr-o linie si apoi le unim intre ele. Asa am facut din toate o componenta impara care are doua noduri cu grad impar. Pentru componentele impare putem sa le conectam folosind atatea muchii cate componente conexe sunt. Conectarea se poate face prin plasarea componentelor ciclic si alegerea a doua noduri de grad impar si folosindu-le pentru a se conecta la componentele vecine. Si acum avem o singura compoenenta conexa si se aplica ideea de mai sus.