

1. U zadatku treba riješiti pojednostavljeni problem pronalazjenja "balansiranih particija" u grafu. Problem se sastoji u sljedećem: Neka je dat graf $G = (V, E)$ sa n čvorova i neka su w_i , $i = 1..n$ težine na čvorovima. Potrebno je odrediti skupove V_1 i V_2 , takve da je $V_1 \cup V_2 = V$, $V_1 \cap V_2 = \emptyset$, tako da je vrijednost $\left| \sum_{i \in V_1} w_i - \sum_{i \in V_2} w_i \right|$ minimalna. Rješenje treba pronaći greedy algoritmom.

- (a) Napraviti sistem za realizaciju grafa. Graf se obavezno realizujekao klasa, a sam način predstavljanja čvorova, grana i težina na čvorovima je proizvoljan.
- (b) Napisati sistem za ispis grafa, tako što se za svaki čvor ispisuje lista njegovih susjeda i njegova težina.
- (c) Realizovati klasu Particija, koja kao obavezne atribute ima dvije liste (može i dva niza) čvorova, koji pripadaju odgovarajućim skupovima V_1 i V_2 , kao i atribute w_1 i w_2 , koji imaju vrijednosti suma težina svih čvorova u V_1 , odnosno u V_2 . Ove vrijednosti se računaju pomoću metoda iz zadatka pod (f). Klasa može da sadrži i dodatne atribute. Sam graf može, ali i ne mora biti atribut. Napisati neophodne metode za funkcionisanje klase. Prilikom kreiranja objekta ove klase, oba skupa V_1 i V_2 su prazni.
- (d) Napisati metod za ispis objekta klase, tako što se ispisuju skupovi V_1 i V_2 , kao i odgovarajuće vrijednosti w_1 i w_2 .
- (e) Napisati metod koji po greedy algoritmu kreira particiju. Algoritam se sastoji od sledećeg:
 - i. U startu su oba skupa V_1 i V_2 prazni, a skup $V' = V$. Iz skupa V' se čvorovi ubacuju u skupove V_1 i V_2 , sve dok V' ne postane prazan. Izbor čvora koji izlazi iz skupa V' je proizvoljan.
 - ii. Čvorovi se ubacuju na sljedeći način: Ako je $w_1 \geq w_2$, naredni čvor se ubacuje u V_2 , u suprotnom u V_1 .
- (f) Napisati metode koji računaju zbir težina čvorova iz V_1 , odnosno u V_2 .
- (g) U glavnom dijelu programa je potrebno kreirati graf, objekat klase Particija, pomoću kog se pravi "balansirana" particija skupa čvorova polaznog grafa. U slučaju uspješne implementacije, pokušati napraviti sistem poboljšanja algoritma, tako što izbor čvora koji izlazi iz skupa V' nije proizvoljan, već zavisi od njegove težine (ispitati da li je bolje uzimati čvor najveće, najmanje težine, ili slučajan čvor).

2. U ML-u napisati funkcije:

- (a) Funkciju $f1$, koja za argument uzima tri liste l, l_1, l_2 i raspoređuje brojeve iz liste l_1 u preostale dvije tako da svi elementi koji se u polaznoj listi javljaju tačno jednom idu u l_1 , a preostali u l_2 .
- (b) Funkciju $f2$ koja za argument uzima listu stringova i formira riječ koja se sastoji samo od onih karaktera koji se, gledajući u svim stringovima ukupno, javljaju tačno jednom.
3. U Prologu napisati relaciju koja za argument uzima tri liste i ispituje da li je zbir elemenata u drugoj i trećoj listi jednak zbiru elemenata iz prve liste.
4. Dokazati korektnost Hartmanovog algoritma.