

Laboratorijska vežba 3 – Algoritmi traženja

Student sa rednim brojem k na spisku u grupi radi zadatak $((k-1) \bmod 20)+1$.

Za izradu funkcija prepoznati i modifikovati odgovarajući algoritam obilaska stabla sa računskih vežbi (dodati nove argumente, promeniti vrednost koju funkcija vraća, izmeniti deo koda algoritma, dodati nove funkcije). Sve probleme rešiti u okviru jednog obilaska grafa, ako nije drugačije navedeno.

Stablo traženja je stablo gde svaki put u stablu odgovara jednom putu u grafu. Unutar puta isti čvor se ne može pojaviti više puta.

Putevi koji nemaju nijedan zajednički čvor osim, eventualno, početnog i krajnjeg čvora su **disjunktni putevi**.

Podgrafovi koji nemaju nijednu granu koja povezuje bilo koji čvor prvog podgrafa sa bilo kojim čvorom drugog podgrafa su **disjunktni podgrafovi**.

1. Napisati funkciju koja određuje visinu stabla traženja (broj nivoa u stablu traženja), za algoritam obilaska grafa po širini, koje se formira za zadati polazni čvor i zadati graf.
2. Napisati funkciju koja određuje broj disjunktnih puteva između dva zadata čvora u grafu. Rešenje ne mora biti optimalno prema broju puteva. Dozvoljeno je više puta pozvati algoritam obilaska grafa.
3. Napisati funkciju koja na osnovu zadatog usmerenog grafa i zadatog (početnog) čvora S formira niz čvorova sa njihovim udaljenostima od čvora S . Udaljenost se određuje kao dužina najkraćeg puta od čvora A do nekog čvora. Dužina puta se određuje kao broj potega koji čine taj put.
4. Napisati funkciju koja na osnovu zadatog neusmerenog grafa i zadatog (ciljnog) čvora G formira neusmereni graf sa heuristikom. Heuristika proizvoljnog čvora C se određuje kao udaljenost čvora C od čvora G . Udaljenost se određuje kao dužina najkraćeg puta između dva čvora. Dužina puta se određuje kao broj potega koji čine taj put.
5. Napisati funkciju koja ispituje koliko disjunktnih podgrafova postoji u zadatom neusmerenom grafu. Dozvoljeno je više puta pozvati algoritam obilaska grafa.
6. Napisati funkciju koja formira stablo traženja za zadati graf, zadati polazni čvor i izabrani algoritam koji se koristiti za obilazak stabla. Student sam bira algoritam za koji se formira stablo traženja.
7. Napisati funkciju koja na osnovu zadatog neusmerenog grafa i dva zadata (ciljna) čvora G_1 i G_2 formira neusmereni graf sa heuristikom. Heuristika proizvoljnog čvora C se određuje kao udaljenost čvora C do bližeg od čvorova G_1 i G_2 . Udaljenost se određuje kao dužina najkraćeg puta između dva čvora. Dužina puta se određuje kao broj potega koji čine taj put. Dozvoljeno je najviše dva puta pozvati prilagođeni algoritam obilaska grafa.

8. Napisati funkciju koja određuje put između zadatog polaznog i ciljnog čvora neusmerenog grafa tako što istovremeno pokreće traženje po širini od polaznog i od ciljnog čvora. Traženje se završava kada se nađe prvi zajednički čvor za oba traženja.
9. Napisati funkciju koja određuje broj čvorova do kojih može da se dođe od zadatog čvora, tako da je dužina puta do čvora jednaka zadatoj vrednosti. Običi samo neophodne čvorove.
10. Napisati funkciju koja pronalazi put u neusmerenom grafu između dva zadata čvora, pri čemu put prolazi kroz treći zadati čvor.
11. Napisati funkciju koja određuje potege koje je moguće izbaciti iz neusmerenog grafa tako da graf i dalje ostane povezan.
12. Napisati funkciju koja pronalazi puteve zadate dužine u neusmerenom grafu između dva zadata čvora.
13. Napisati funkciju koja određuje listu grana u grafu koje je neophodno obrisati da bi se ciklični graf transformisao u aciklični (da bi se u njemu eliminisali ciklusi).
14. Napisati funkciju koja određuje da li je usmereni graf jako povezan graf (da li postoji put između bilo koja dva čvora u grafu).
15. Napisati funkciju koja proverava da li graf sadrži Ojlerov put (put kojim se obilazi celokupan graf, uz obilazak svake grane tačno jednom).
16. Napisati funkciju koja na osnovu zadatog usmerenog grafa i zadata tri čvora S, M i G određuje put (takav da ni kroz jedan čvor ne prolazi dva puta) od čvora S do čvora G preko čvora M. Dozvoljeno je najviše dva puta pozvati prilagođeni algoritam obilaska grafa.
17. Napisati funkciju koja na osnovu zadatog težinskog neusmerenog grafa i zadatog (ciljnog) čvora G formira neusmereni težinski graf sa heuristikom. Heuristika proizvoljnog čvora C se određuje kao dužina puta od čvora C do čvora G.
18. Napisati funkciju koja na osnovu zadatog usmerenog grafa i zadatog čvora S određuje čvor (čvorove) koji su najudaljeniji od njega. Udaljenost se određuje kao dužina najkraćeg puta između dva čvora. Dužina puta se određuje kao broj potega koji čine taj put.
19. Napisati funkciju koja na osnovu zadatog usmerenog grafa i zadata dva čvora S i M određuje ciklični put (takav da ni kroz jedan čvor ne prolazi dva puta) koji počinje i završava se u čvoru S i prolazi kroz čvor M. Dozvoljeno je najviše dva puta pozvati prilagođeni algoritam obilaska grafa.
20. Napisati funkciju koja na osnovu zadatog težinskog neusmerenog grafa i dva zadata (ciljna) čvora G1 i G2 formira težinski graf sa heuristikom. Heuristika proizvoljnog čvora C se određuje kao dužina puta od čvora C do bližeg od čvorova G1 i G2. Dozvoljeno je najviše dva puta pozvati prilagođeni algoritam obilaska grafa.