ALGORITMI NA GRAFOVIMA

PROBLEM

Po urbanističkom planu, dva grada su međusobno povezana određenim skupom kablova različite namene, međutim nisu svaka dva grada direktno povezana. Ako se dva grada ne mogu direktno povezati, njihova veza se ostvaruje preko grada koji jeste direktno povezan sa njima. Telekomunikaciona kompanija želi da zakupi neke od tih kablova kako bi počela da radi na tom tržištu, a kako cena zakupa zavisi od dužine kablova, želi da je zakup najmanje košta.

Problem se može predstaviti grafom gde čvorovi grafa predstavljaju gradove a kablovi grane koje dva čvora povezuju. Telekomunikaciona kompanija bi svoje sedište postavila u jednom gradu i do ostalih gradova došla iz svog sedišta po najkaćem putu. Odnosno, jezikom teorije grafova, želimo da napravimo minimalni razapinjujući podgraf datog težinskog grafa sa n čvorova.

Pomenuti problem predstavlja problem najmanjeg razapinjujućeg stabla (eng. Minimum Spanning Tree). U okviru ovog kursa biće predstavljena dva algoritm za rešavanje MST problema koji spadaju u proždrljive algoritme (eng. greedy algorithms)

- Kruskalov algoritam
- Primov algoritam

Postupak određivanja minimalnog razapinjujućeg stabla može se posmatrati i na sledeći način:

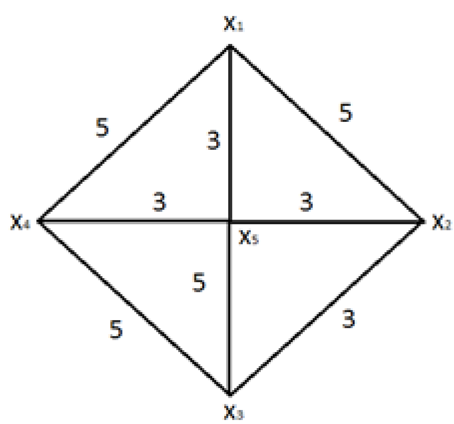
Neka je T razapinjujuće stablo nad nekim grafom G i neka su T_1 i T_2 bilo koja dva poddrveta drveta T dobijena raskidanjem njihove trenutne veze. Posmatramo skup svih veza između T_1 i T_2 i povezujemo ih granom

$$\Delta_{ij} = \min_{x_i \in T_1}\{\min_{x_j \in T_2}\{c(x_i, x_j)\}\}.$$

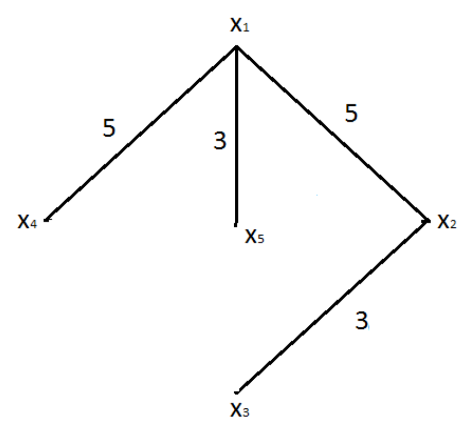
Ponavljanjem ovog postupka može se dobiti najmanje razapinjujuće stablo.

PRIMER 76

Naći najmanje razapinjujuće stablo za sledeći graf

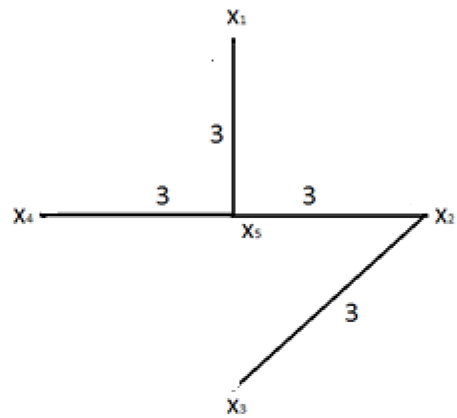


Rešenje



Prvo ćemo napraviti drvo čiji je koren npr. čvor x_1.

Dobijeno drvo je težine 18. Naš zadatak je da napravimo drvo najmanje težine. Za početak, ako razdvojimo čvor x_4 od drveta, odnosno prekinemo granu x_1x_4 primetićemo da će se jeftinija veza sa tim čvorom ostvariti preko čvora x_5, odnosno preko grane x_5x_4. Slično, veza između čvora x_2 i x_5 je jefinija nego preko nekog posrednog čvora pa ćemo i to zameniti. Konačno, najmanje razapinjujuće stablo ima sledeći oblik:



Postoji više načina za dobijanje najmanjeg razapinjujućeg stabla, mi ćemo posmatrati Kruskalov algoritam i Primov algoritam.

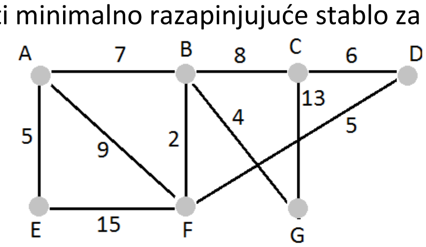
Kruskalov algoritam za nalaženje najmanjeg razapinjujućeg stabla

Razapinjujuće tablo minimalne težine u težinskom grafu G naziva se *opzimalnim stablom.* Za konstrukciju optimalnog stabla koristimo tzv. Kruskalov algoritam koji se može opisati na sledeći način:

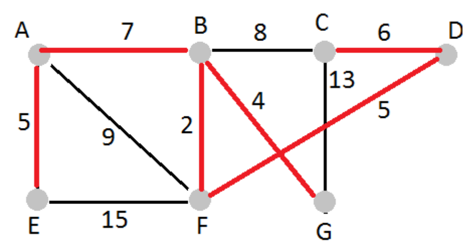
KORAK1 Početi od potpuno nepovezanog grafa T od n čvorova.
KORAK2 Poređati sve grane grafa G tako da im težina bude u rastućem redosledu
KORAK3 Počevši od najjeftinije grane dodavati ih grafu T vodeći računa da se ne napravi cikl.
KORAK4 Ponoviti KORAK3 sve dok graf T ne bude imao $n-1$ granu.
Dobijen podgraf T predstavlja minimalno razapinjujuće stablo grafa G.

TEOREMA: Svako razapinjujuće stablo konstruisano Kruskalovim algoritmom je optimalno.

PRIMER 77
Primernom Kruskalovog algoritma odrediti minimalno razapinjujuće stablo za sledeći graf:

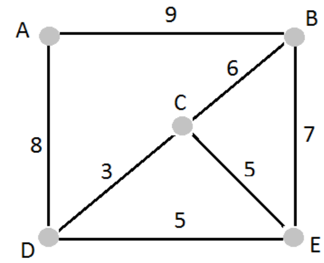


REŠENJE:
Odgovarajuće tablo je sledećeg oblika:



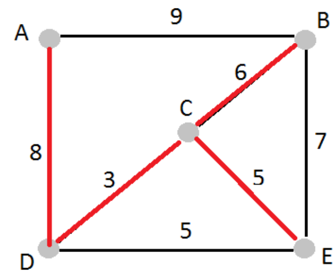
PRIMER 78

Primernom Kruskalovog algoritma odrediti minimalno razapinjujuće stablo za sledeći graf:



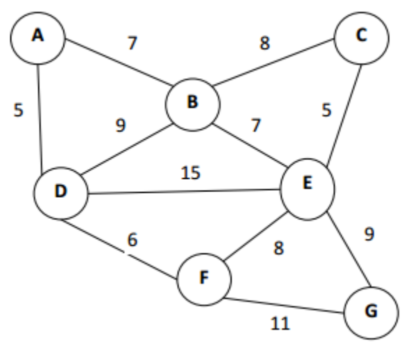
REŠENJE:

Odgovarajuće tablo je sledećeg oblika:



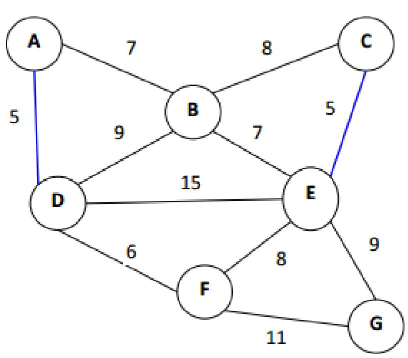
PRIMER 79

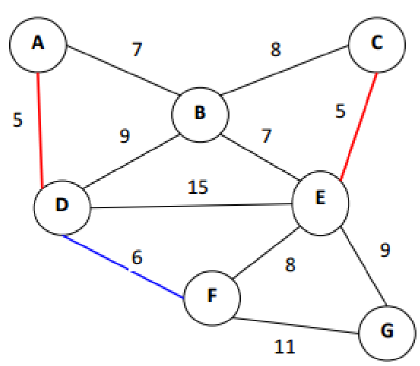
Primernom Kruskalovog algoritma odrediti minimalno razapinjujuće stablo za graf dat slikom:

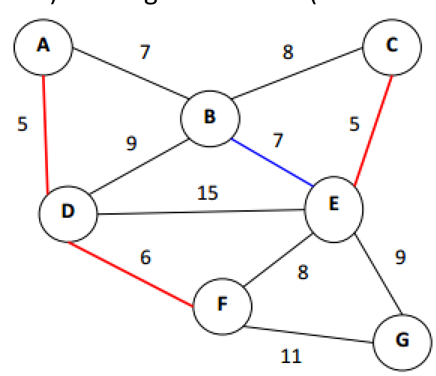


REŠENJE

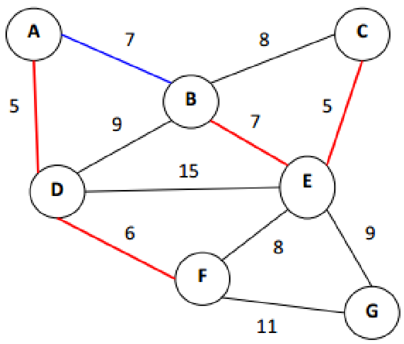
Na početku, drvo T se sastoji se samo iz skupa čvorova $\{A, B, C, D, E, F, G\}$. Primetimo da su najjeftinije grane A-D i C-E, dodajmo ih drvetu (slika levo). Sledeće grane po veličini su DF (slika u sredini). Slede grane AB i BE (slika desno)

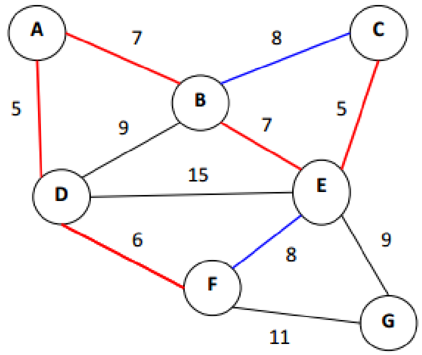


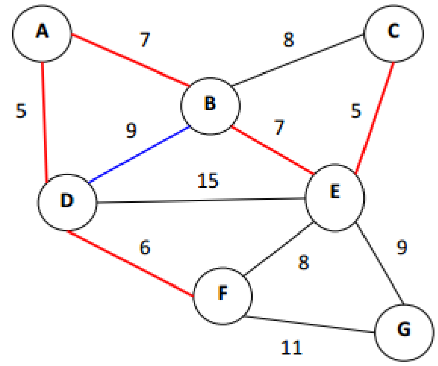


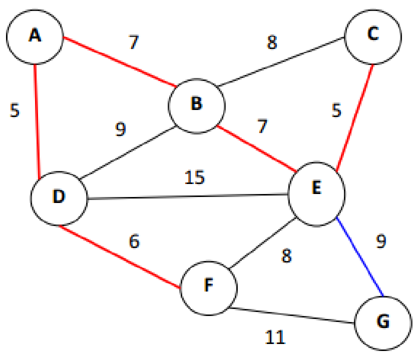


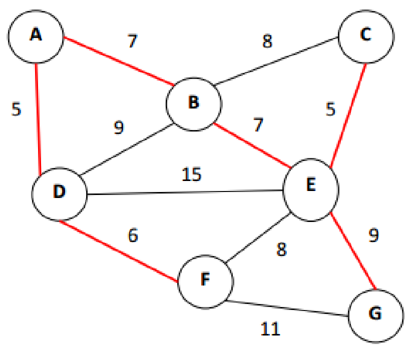
Sledeće grane po veličini su AB, BC i EF. Biramo granu AB s'tim da dodavanje grane BC nije dozvoljeno obzirom da bismo dobili cikl što nije dozvoljeno, tako da se ta grana ne može dalje uvrstiti u graf. Takođe, ni dodavanje grane EF nije dozvoljeno. Sledeća po veličini grana koju možemo je BD koju takođe ne možemo dodati. Dodajemo granu EG











Primov algoritam za nalaženje najmanjeg razapinjujućeg stabla

Primov algoritam za nalaženje minimalnog razapinjujućeg stabla može se konstruisati na sledeći način:

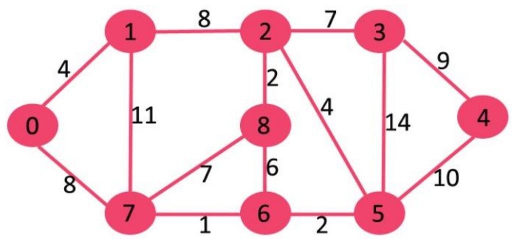
KORAK1 Proizvoljno uzeti nerazmatrani čvor (ako postoj) za tekuću komponentu. Na početku programa svi čvorovi se smatraju nerazmatranim. Ako takav čvor ne postoji preći na KORAK3 u protivnom izabrani čvor markirati kao razmazran.

KORAK2 Naći najlakšu granu koja povezuje čvor iz tekuće komponente sa nekim čvorom van nje. Ako takva grana ne postoji, preći na KORAK1. U protivnom, tu granu priključiti tekućoj komponenti kao i onaj krajnji čvor grane koji nije bio u komponenti. Ujedno markirati da je dotični čvor razmatran. Preći na početak KORAKA2.

KORAK3 Završiti algoritam.

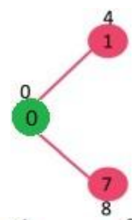
PRIMER 80

Odrediti minimalno razapinjujuće stablo za sledeći graf



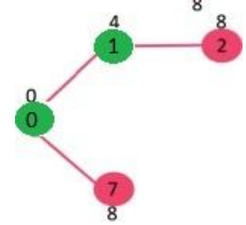
Rešenje

Obeležimo tekući skup sa $T = \{0, \infty, \infty, \dots, \infty\}$ gde ∞ predstavlja marker da veza sa odgovarajućim čvorom nije uspostavljena. Biramo čvor minimalnog rastojanja od čvora 0:



Na raspolaganju su nam čvorovi 1 i 7 čija odgovarajuća rastojanja iznose 4 i 8 (tim redom). Zelenom bojom obeležimo čvor koji smo markirali.Biramo čvor sa najmanjom težinom koji se do sada nije uključio u drvo T. Dakle, dodajemo čvor 1. Sada je drvo obliks $T = \{0,1\}$.

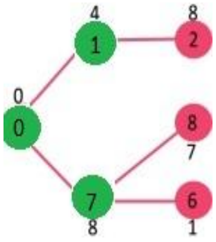
Čvor 1 povezan je čvorovima 2 I 7 od kojih je čvor 7 već jednom razmatran. Dakle, ili ćemo dodati čvor 7 koji je za 8 udaljen od početnog čvora ili čvor 2 koji j za 8 udaljen od čvora 1.

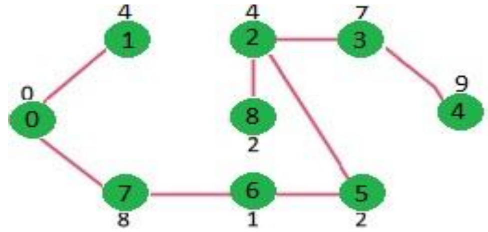


Novo stablo ima sledeći oblik $T = \{0,1,2\}$. Čvor 1 obeležimo kao već razmatrani čvor.

Dalje posmatramo odstojanje od čvorova 1 i 7. Primetimo da su nova rastojanja 8, 7 i 1. Proširujemo naše drvo dodavanjem čvora 6. $T = \{0, 1, 7, 6\}$.

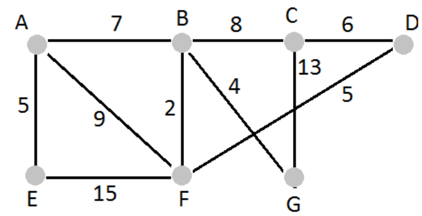
Novi čvorovi za razmatranje su 2,8 i 6 (na rastojanju 8,6,2). Buramo čvor 6 i tako redom. Konačno, traženo stablo je oblika:





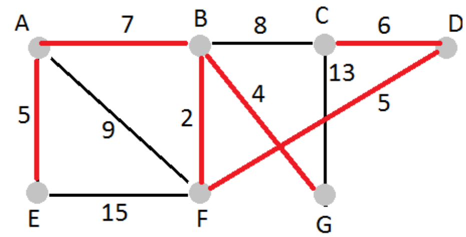
PRIMER 81

Primernom Primovog algoritma odrediti minimalno razapinjujuće stablo za sledeći graf:



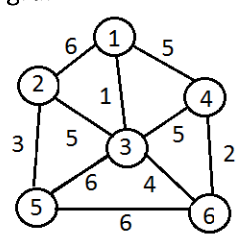
REŠENJE:

Odgovarajuće tablo je sledećeg oblika:



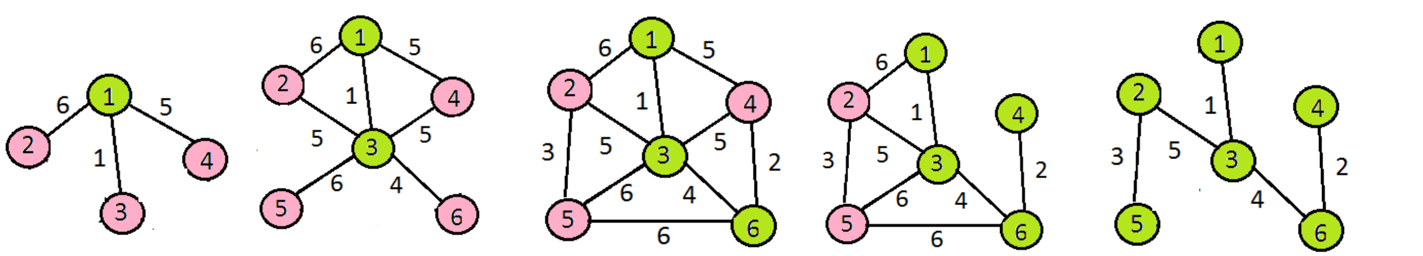
PRIMER 82

Odrediti minimalno razapinjujuće stablo za sledeći graf



Rešenje:

Biramo čvor 1 kao početni. Rastojanje od čvora 1 do svih ostalih čvorova iznosi $\{6,1,5,\infty,\infty,\infty\}$. Biramo čvor 3 kao sledeći. Dakle $T=\{1,3\}$. Rastojanje stabla T od preostalih čvorova sada iznosi $\{5,5,6,4\}$. Biramo čvor 6 kao naredni...



Algoritam Dijkstre za nalaženje najkraćeg rastojanja između dva čvora

Algoritam Dijkstre je razvijen 1959. i smatra se jednim od najefikasnijih i najpogodnijih algoritama za određivanje najkraćeg puta između dva zadata čvora. Neophodan uslov za primenu ovog algoritma je da je dužina c_{ij} svake grane $(i,j) \in L$ nenegativna.

Prva tri koraka određuju dužinu najkraćeg puta dok poslednji korak identifikuje traženi put

KORAK 1

Svim čvorovima mreže dodeliti sledeće oznake: čvoru s dodeliti oznaku $d^+(s) = 0$ koju ćemo smatrati nepromenljivom, a svim ostalim čvorovima, uključujući i čvor $\boldsymbol{t}$, dodeliti promenljive oznake $d^-(i) = -\infty$ a zatim preći na KORAK2.

KORAK 2

Staviti $s = i$, a zatim odrediti skup čvorova $j \in A_i$ sa promenljivim oznakama gde A_i predstavlja skup čvorova sledbenika čvora i. Za takve čvorove$j$ odrediti nove, takođe promenljive oznake na sledeći način:

$$d(j) = \min\{d^-(j), d(i) + c_{ij}\} \qquad (1)$$

Od svih novodobijenih oznaka samo jedna oznaka je nepromenljiva i određuje se iz uslova

$$d^+(j) = min\{d(j)\} \qquad (2)$$

Dok su sve ostale promenljive. Nakon određivanja ovih oznaka preći na KORAK 3.

KORAK 3

Proveriti da li je čvor $\boldsymbol{t}$ označen nepromenljivom oznakom i u zavisnosti od toga postoje sledeće dve mogućnosti:

K 3.1: Čvor $\boldsymbol{t}$ je označen nepromenljivom oznakom $d^+(t)$, koja u stvari predstavlja dužinu traženog najkraćeg puta, pa se postupak obeležavanja čvorova završava i potrebno je preći na postupak izdvajanja najkraćeg pita, odnosno preći na korak4.

K 3.2: Čvor $\boldsymbol{t}$ nije označen nepromenljivom oznakom. Staviti da je $i = j$, gde je $\boldsymbol{j}$ čvor koji u prethodnom postupku dobio nepromenljivu oznaku $d^+(j)$ i vratiti se na korak3.

KORAK 4

Ovim korakom se izdvaja najkraći put. Postupak počinjemo iz čvora $\boldsymbol{t}$. U poslednjoj iteraciji čvoru $\boldsymbol{t}$ dodeljena je neporomenljiva oznaka $d^+(t)$ dok su u prethodnim iteracijama preostalim čvorovima mreže $i \in N$ dodeljene ili promenljive oznake $d^-(i)$ ili nepromenljive oznake $d^+(i)$. Od svih čvorova susednih čvoru $\boldsymbol{t}$ odredimo čvor j_k sa nepromenljivom oznakom $d^+(j_k)$ za koji je

$$d^+(t) - d^+(j_k) = c_{j_k t}\,. \qquad (3)$$

Vidimo da čvor j_k sigurno postoji, jer se pomoću takvog čvora i određuje nepromenljiva oznaka čvora $\boldsymbol{t}$. Isto tako je jasno da za sve ostale čvorove susedne čvoru $\boldsymbol{t}$, koje možemo označiti sa q_r i koji mogu imati promenljive ili nepromenljive oznake, važi

$$d^+(t) - d(q_r) \le c_{q,t} \quad . \qquad (4)$$

Ponavljanjem ovog postupka, to jest određivanjem čvora j_{k-1}, susednog čvoru j_k, koji ima nepromenljivu oznaku $d^+(j_{k-1})$, za koji je

$$d^+(j_k) - d^+(j_{k-1}) = c_{j_{k-1},j_k}, \qquad (5)$$

Na kraju se dolazi do čvora j_1, kome je susedni čvor $\boldsymbol{s}$, i za koji je

$$d^+(j_1) - d^+(s) = c_{s_{j_1}}. \qquad (6)$$

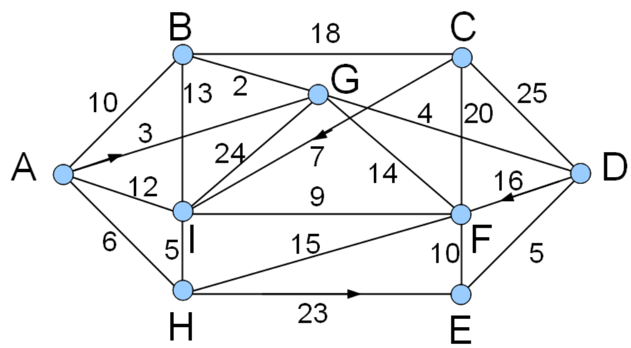
Sada smo dobili put koji ide preko čvorova $s, j_1, j_2, \dots, j_{k-1}, j_k, t$. Sabiranjem relacija (3), (5) i (6) za sve čvorove na tom putu (vodimo računa o tome da je $d(s) = 0$) dobijamo da je dužina puta jednaka nepromenljivoj oznaci čvora ***t***, to jest dobijamo da je dužina puta

$$d^{+}(t) = c_{s_{j_1}} + c_{j_1 j_2} + \cdots + c_{j_k t} \qquad (7)$$

Dobijeni put je i najkraći put između zadataih čvorova ***s*** i ***t***.

PRIMER 83

Koristeći algoritam Dijkstra-e naći najkraći put između čvorova A i bilo kog drugog čvora date mreže.



- Mreža može imati više najkraćih puteva između dva čvora i to se može uočiti u toku same postavke algoritma (postoji više čvorova koji zadovoljavaju relaciju 2).

Modifikacija algoritma Dijkstra-e

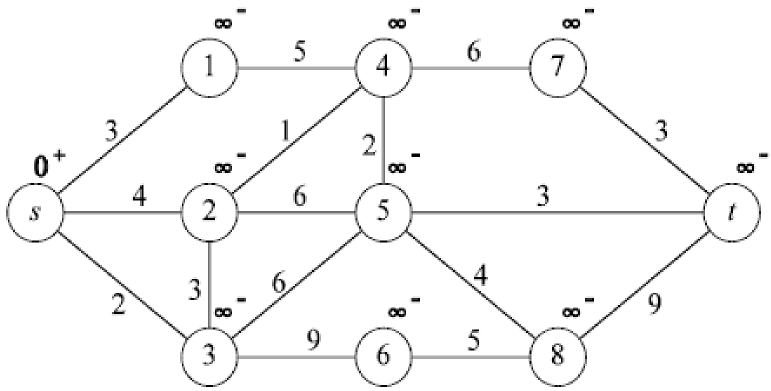
- Prva dva koraka se spoje i prilikom određivanja oznaka čvorova pored svake nepromenljive oznake se upisuje i indeks čvora na osnovu kojeg je, u prethodnom koraku, dobijena ta oznaka, onda se time može i rešiti i problem izdvajanja najkraćeg puta. Dakle, svaki čvor $j \in N$ i $j \neq s$ umesto ranije nepromenjene oznake $d^{+}(j)$ sada bi imao isto tako nepromenljivu, oznaku koja se, po konvenciji, može pisati na jedan od sledećih načina:

  $[i, d^{+}(j)]$ ili $[d^{+}(j), i]$ ili $[i, d(j)]$ ili $[d(j), i]$.

  gde je ***i*** čvor susedni čvoru ***j*** pomoću kojeg se, primenom relacije (7) određuje oznaka $d^{+}(j)$. Na ovaj način bi čvor ***t*** dobio novu oznaku, recimo $[d(t), j_k]$, što znači da najkraći put do čvora ***t*** ide preko čvora j_k. Čvor je već ranije označen novom oznakom $[d(j_k), j_{k-1}]$, što znači da najkraći put ide i preko čvora j_{k-1}. Produžavajući ovaj postupak na kraju se dolazi do čvora ***s***. Ovim postupkom se problem izdvajanja najkraćeg puta jednostavno rešava, bez ikakvih dopunskih izračunavanja.

PRIMER 84 (preuzeto iz literature sa FONa)

Koristeći Dijkstrin algoritam odrediti najkraće rastojanje od čvora s do čvora t.



Rešenje:

Zadata mreža sa početnim oznakama je data na slici. Prelazimo na iterativni deo primene obeležja.
Sitna obeležja su data na slici.

1) Čvor s slede čvorovi: 1, 2 i 3 → $A_s = \{1,2,3\}$

$$d^-(1) = \min\{0+3, \infty\} = 3$$
$$d^-(2) = \min\{0+4, \infty\} = 4$$
$$d^-(3) = \min\{0+2, \infty\} = 2$$
$$\min\{d^-(1), d^-(2), d^-(3)\} = d^-(3) = 2 \rightarrow d^+(3) = d^-(3).$$

2) Čvor 3 slede čvorovi: 2, 5 i 6 → $A_3 = \{2,5,6\}$

$$d^-(2) = \min\{2+3, \infty\} = 4$$
$$d^-(5) = \min\{5+6, \infty\} = 8$$
$$d^-(6) = \min\{6+9, \infty\} = 11$$
$$\min\{d^-(1), d^-(2), d^-(5), d^-(6)\}, = d^-(1) = 3 \rightarrow d^+(1) = d^-(1).$$

3) $A_1 = \{4\}$

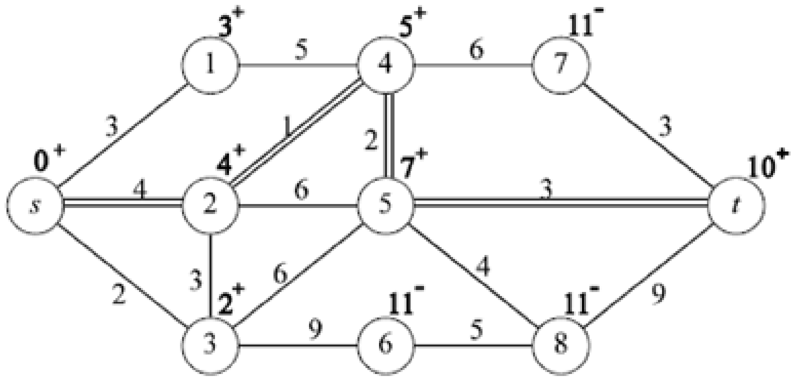
$$d^-(4) = \min\{3+5, \infty\} = 8$$
$$\min\{\, d^-(2), d^-(4), d^-(5), d^-(6)\}, = d^-(2) = 4 \rightarrow d^+(2) = d^-(2).$$

4) A_2

$$d^-(4) = \min\{4+1{,}8\} = 5$$
$$d^-(5) = \min\{4+6{,}8\} = 8$$
$$\min\{\, d^-(4), d^-(5), d^-(6)\}, = d^-(4) = 5 \rightarrow d^+(4) = d^-(4).$$

5) $A_4 = \{5,7\}$

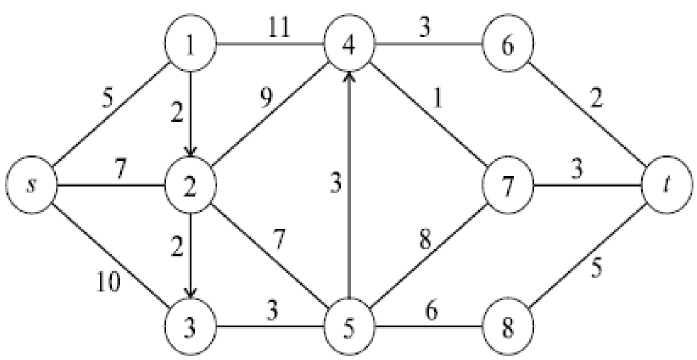
$$d^-(5) = \min\{5+2{,}8\} = 7$$
$$d^-(7) = \min\{5+6, \infty\} = 11$$
$$\min\{\;\, d^-(5), d^-(6), d^-(7)\}, = d^-(5) = 7 \rightarrow d^+(5) = d^-(5).$$



Prelazimo na određivanje najkraćeg puta
$10 - 3 = 7 \;\rightarrow\;$ čvor 5
$7 - 2 = 5 \;\rightarrow\;$ čvor 4
$5 - 1 = 4 \;\rightarrow\;$ čvor 2 i
$4 - 4 = 0 \;\rightarrow\;$ vratili smo se na početni čvor s.
Najkraći put u mreži je $(s, 2,4,5, t)$
Korisno je proveriti da li dužina dobijenog puta odgovara obeležju $d^{+}(t) = 3 + 2 + 1 + 4 = 10$

Potrebno je napomenuti da stalno obeležje čvora predstavlja dužinu najkaćeg puta od čvora s do tog čvora. Kada se rešava problem određivanja najkraćeg puta između početnog i svih ostalih čvorova u mreži, potrebno je produžiti postupak obeležavanja dok svi čvorovi ne dobiju stalna obeležja.

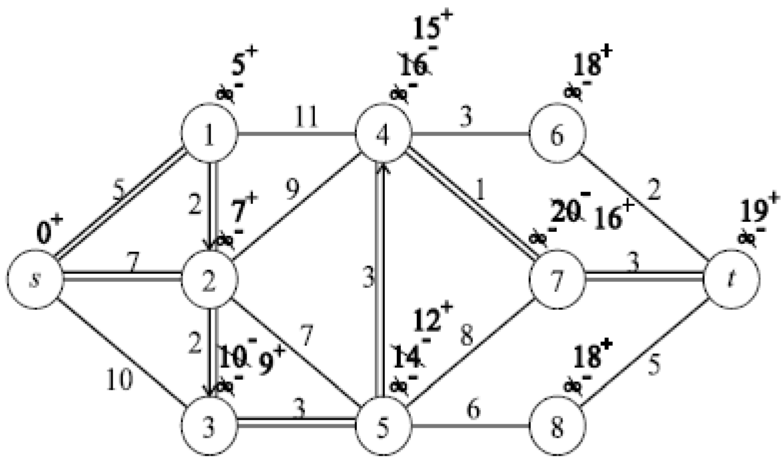
PRIMER 85 (preuzeto iz literature sa FONa)



a) Odrediti najkraći put između čvorova s i t i njegovu dužinu koristeći Dijkstrin algoritam.
b) Odrediti dužine najkraćih puteva između čvora s i svih preostalih čvorova u mreži.

Rešenje

a) Zadatak ćemo rešiti na samoj mreži:



Slika 1.6.

Dužina najkraćeg puta je 19. Kod određivanja puta otkrivamo da je rešenje višestruko jer u čvoru 2 postoje dva prethodna čvora koja zadovoljavaju uslov: $d^+(2) - d^+(j) = c_{j2}$, gde j predstavlja susedne čvorove čvora 2. Ti putevi su:

I : $(s, 1, 2, 3, 5, 4, 7, t)$

II : $(s, 2, 3, 5, 4, 7, t)$.

b) Ovaj deo zadatka je implicitno već rešen u postupku pod (a).

Vidimo da su svi čvorovi na mreži (slika 1.6.) obeleženi stalnim obeležjima koja predstavljaju dužine najkraćih puteva od čvora s do njih. Dakle, odgovarajuće dužine iznose:

s - 1	: 5	s - 5	: 12	s - t	: 19
s - 2	: 7	s - 6	: 18		
s - 3	: 9	s - 7	: 16		
s - 4	: 15	s - 8	: 18		

Pošto se pod (b) tražilo da se odrede samo dužine puteva, ovim je zadatak rešen. ■

Belman-ov algoritam

- modifikovana verzija – pamte se indeksi čvorova kroz koji prolazi najkraći put

KORAK 1

Staviti da je $k = 1$ gde *k* predstavlja redni broj iteracije, i $r = s$ a zatim odrediti skup *R* čvorova do kojih se može doći iz čvora ***r*** po nekoj grani. Čvorovima mreže dodeliti oznake:

$$d^1(r) = 0 \; za \; r = s$$
$$d^1(i) = c_{si} \; za \; i \in R$$
$$d^1(i) = \infty \; za \; i \notin R \; i \; i \neq s$$

A zatim preći na KORAK2.

KORAK 2

Neka su *k*-toj iteraciji svi čvorvi $i \in N$ označeni sa $d_e^k(i)$, gde je sa ***e*** označen čvor na osnovu kojeg je određena oznaka $d_e^k(i)$ onda se u *k+*1-oj iteraciji oznake menjaju i određuju iz sledećeg uslova:

$$d_e^{k+1}(i) = \min\{d_e^k(i), \min_j[d_n^k(j) + c_{ij}\}$$

gde je $j \in B_i \cap R$ a B_i predstavlja skup čvorova koji prethode čvoru ***i***, dok su sa ***g*** i ***h*** označeni čvorovi na osnovu kojih su određene oznake $d_g^{k+1}(i)$ i $d_n^k(i)$. Preći na korak3.

KORAK 3

K 3.1: Ako se za neko $k \leq n$ (*n* broj čvorova digrafa) dobije da je za svako $i \in N, d_g^{k+1}(i) = d_e^k(i)$, tada je dužina najkraćeg traženog puta između čvorova *s* i *t* određena, a $d_g^k(i)$ predstavlja dužinu tog puta, gde je sa *q* označen broj čvora na osnovu kojeg je dobijena osznaka $d_g^k(t)$. Preći na korak4.

K 3.2: Ako se za neko $k < n$ dobije da je $d_g^{k+1}(i) \neq d_e^k(i)$, samo za neke čvorove $i \in N$, tada formiramo novi skup *R* na sledeći način:

$$R = \{i | d_g^{k+1}(i) \neq d_e^k(i)\}$$

A zatim stavimo da je $k = k + 1$ i idemo na korak2.

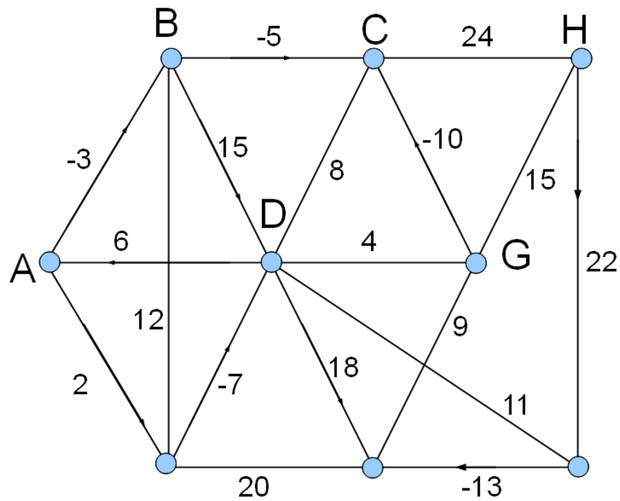
K 3.3: Ako se za $k = n$ dobije da je $d_g^{k+1}(i) \neq d_e^k(i)$, samo za neke čvorove $i \in N$ tada postoji kontura negativne dužine i ne postoji rešenje postavljenog problema, čime se algoritam završava.

KORAK 4

U ovom koraku se idfentifikuje najkraći put između zadatih čvorova ***s*** i ***t***. Obzirom da je u prethodnom koraku određena veličina pored koje je upisan čvor ***q***, na osnovu kojeg je izračunata, to znači da najkraći put do čvora ***t*** vodi preko čvora ***q***. Čvor ***q*** je na osnovu neke od prethodnih iteracija takođe označen, i pored te oznake je upisan čvor na osnovu koga je izračunata ta oznaka, što znači da i taj čvor pripada najkraćem putu između čvorova ***s*** i ***t***. Produžavajući postupak dobijamo niz čvorova koji čini najkraći put.

- Bellman-ov algoritam predstavlja modifikaciju algoritma Dijkstra-e.
- Ovim algoritmom se dobija najkraći put od jednog do svih preostalih čvorova date mreže.

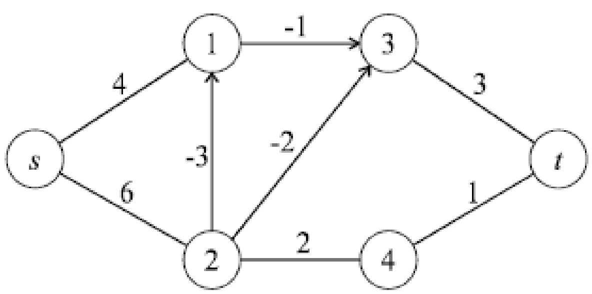
PRIMER 86



Primer je uzet iz knjige Graph theory.

PRIMER 87 (preuzeto iz literature sa FONa)

Odrediti najkraći put i njegovu dužinu za čvorove s i t mreže prikazane na slici 1.7.



Slika 1.7.

Rešenje:

$k = 1, R = \{1, 2\}$

$d^1(s) = 0$	$d^1(3) = \infty$
$d^1(1) = 4$	$d^1(4) = \infty$
$d^1(2) = 6$	$d^1(t) = \infty$

Čvorovi koji slede skup čvorova R su: $\Gamma(R) = \{s, 1, 3, 4\}$ i za njih računamo nove oznake:

$d^2(s) = \min\{0, \min[4+4, 6+6]\} = 0$

$d^2(1) = \min\{4, \min[0+4, 6\text{-}3]\} = \min\{4, 3\} = 3$ ☜

$d^2(3) = \min\{\infty, 4\text{-}1, 6\text{-}2\} = 3$ ☜

$d^2(4) = \min\{\infty, 6+2\} = 8$ ☜

Pošto je došlo do promene bar jednog $d^{k-1}(i)$ (označeni sa „☜“), nastavljamo sa iteracijama.

$k = 2$, $R = \{1, 3, 4\}$
$\Gamma(R) = \{s, 2, 3, t\}$
$d^3(s) = \min\{0, 4+4, 6+6\} = 0$
$d^3(2) = \min\{6, 8+2\} = 6$
$d^3(3) = \min\{3, 6\text{-}2, 3\text{-}1\} = 2$ ☜
$d^3(t) = \min\{\infty, 3+3, 8+1\} = 6$ ☜

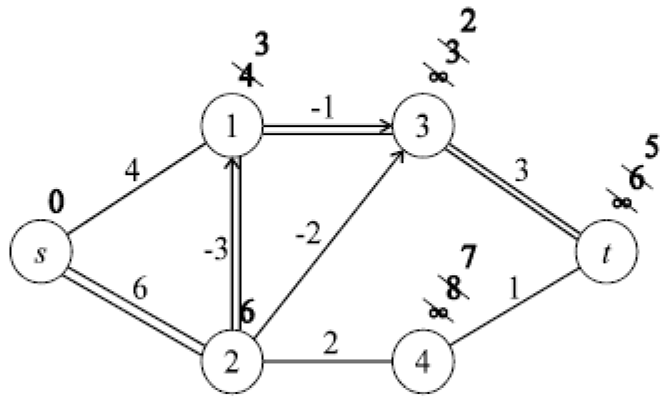
$k = 3$, $R = \{3, t\}$
$\Gamma(R) = \{3, 4, t\}$
$d^4(3) = \min\{2, 6\text{-}2, 3\text{-}1, 6+3\} = 2$
$d^4(4) = \min\{8, 6+2, 6+1\} = 7$ ☜
$d^4(t) = \min\{6, 2+3, 8+1\} = 5$ ☜

$k = 4$, $R = \{4, t\}$
$i \in \{2, 3, 4, t\}$
$d^5(2) = \min\{6, 7+2\} = 6$
$d^5(3) = \min\{2, 6\text{-}2, 3\text{-}1, 5+3\} = 2$
$d^5(4) = \min\{7, 6+2, 5+1\} = 6$ ☜
$d^5(t) = \min\{5, 2+3, 8+1\} = 5$

$k = 5$, $R = \{4\}$
$i \in \{2, t\}$
$d^6(2) = \min\{6, 7+2\} = 6$
$d^6(t) = \min\{5, 2+3, 7+1\} = 5$

Pošto u poslednjoj iteraciji nije promenjeno ni jedno obeležje, a k je manje od $n = 6$, zaključujemo da je postupak završen. Dužina najkraćeg puta od čvora s do t iznosi 5, a put je: (s, 2, 1, 3, t). Poslednje obeležje čvora predstavlja dužinu najkraćeg puta od čvora s do tog čvora.

Postupak određivanja puta i njegove dužine se može sprovesti i na samoj mreži:



Slika 1.8.

■

Određivanje najkraćih puteva između svaka dva čvora date mreže

- Problem određivanja najkraćeg puta između svaka dva čvora date mreže može da se reši primenom algoritma Dijkstra-e ili primernom Bellmanovog algoritma kroz *n* ponavljanja (*n* – broj čvorova mreže) ali bi tada broj računskih operacija bio veliki. Iz ovog razloga razvili su se neki drugi algoritmi, jedan od njih je Floyd-ov algoritam.
- Primenom Floyd-ovog algoritma mogu se odrediti samo dužine najkraćih puteva između svih parova čvorova date mreže ali se ne dobijaju informacije o idetifikaciji puteva. Dakle, dobija se dužina najkraćeg puta ali ne i sam put.
- Modifikacijom Floyd-ovog algoritma dobijamo Floyd-Warshall-ov algoritam pomoću koga se pored oderđivanja dužine najkraćeg puta identifikuje i sam put.

- Koristimo sledeće oznake:
    - Dužine puteva uzmeđu dva čvora su realni brojevi $c_{ij}, (i,j) \in L$
    - $c_{ij} = 0, i \in N$
    - $c_{ij} = \infty, (i,j) \notin L$

FLOYD-OV ALGORITAM ZA NALAŽENJE NAJKRAĆEG RASTOJANJA IZMEĐU SVAKA DVA ČVORA

KORAK 1

Posmatramo matricu $C^0 = [c_{ij}^0]$ na sledeći način:

$$c_{ij}^0 = c_{ij}, (i,j) \in N$$
$$c_{ij} = \infty, (i,j) \notin L$$
$$c_{ij} = 0, i \in N$$

Na osnovu matrice $C^0 = [c_{ij}^0]$ formiramo zatim matricu $D^0 = [d_{ij}^0]$ čiji su elementi

$$d_{ij}^0 = j \quad \text{ako je} \quad c_{ij}^0 \neq \infty$$
$$d_{ij}^0 = 0 \quad \text{ako je} \quad c_{ij}^0 = \infty$$
$$d_{ij} = 0, \qquad i \in N$$

Preći na KORAK2.

KORAK 2

Staviti da je $k = 1$ (u opštem slučaju $k = k + 1$) i odredtiti skupove čvorova I i J na sledeći način:

$$I = \{i | i \neq k, c_{ij}^0 \neq \infty\}$$
$$J = \{j \big| j \neq k, c_{kj}^0 \neq \infty\}$$

a zatim za svako $i \in I$ i $j \in J$ izračunati elemente matrice C^{k+1} pomoću sledeće relacije:

$$c_{ij}^{k+1} = \min\left\{c_{ij}^k, c_{i,k+1}^k + c_{k+1,j}^k\right\}$$

dok su svi preostali elememnti matrice C^{k+1} jednaki odgovarajućim elementima matrice C^k.

Na osnovu matrica C^{k+1} i C^k određuju se i elementi matrice D^{k+1} na sledeći način

$$d_{ij}^{k+1} = k + 1 \text{ ako je } c_{ij}^{k+1} \neq c_{ij}^k$$
$$d_{ij}^{k+1} = d_{ij}^k \text{ ako je } c_{ij}^{k+1} = c_{ij}^k$$

Postupak određivanja matrica $C^{k+1} = [c_{ij}^{k+1}]$ i $D^{k+1} = [d_{ij}^{k+1}]$ se završava. Prelazi se na KORAK 3.

KORAK 3

Proveriti da li je $k \leq n$ a zatim preći na korake K 3.1-K 3.3.

K 3.1: Ako se za neko $k < n$ i ako su svi $c_{1i}^{k} \leq 0$ staviti da je $k = k + 1$ i ići na korak 2.

K 3.2: Ako se za bilo koje $k \leq n$ dobije da je $c_{ii}^{k} < 0$, to znači da postoji kontura negativne dužine koja sadrži čvor ***i***. Ako se ovaj slučaj dogodi sledi da rešenje ne postoji i algoritam se završava.

K 3.3: Ako se ustanovi da je $k = n$ i da su svi $c_{ii}^{n} > 0$, tada se proces izračunavanja i određivanja elemenata matrice C^k i D^k završava, a elementi c_{ij}^{n}, matrice C^n predstavljaju tražene dužine najkraćih puteva između svaka dva čvora date mreže, dok se na osnovu elemenata d_{ij}^{n} matrice D^n vrši indentifikacija tih puteva pto se obavlja u koraku br4.

KORAK 4

U ovom koraku se identifikuje najkraće puteve između čvorova $\boldsymbol{i}$ i $\boldsymbol{j}$ čije su dužine date u matrici C^n:

U posebnu tabelu upišemo broj jednak indeksu $\boldsymbol{j}$ a zatim uočimo element d_{ij}^{n} matrice D^n. Neka je taj element jednak nekom broju s_1. Ovaj broj upišemo u tabelu ispred broja jednakog indeksu ***j***, a zatim proveravamo da li je broj s_1 jednak indeksu $\boldsymbol{i}$. U zavisnosti od toga imamo dve mogućnosti:

K 4.1: Ako je $s_1 = i$ tada je postupak identifikacije najkraćeg puta između čvorova $\boldsymbol{i}$ i $\boldsymbol{j}$ završen, a čvorovi preko kojih ide taj put su dati u tabeli i time se algoritam završava.

K 4.2: Ako je $s_1 \neq i$ onda uočimo element $d_{1s_1}^{n}$, matrice D^n, i neka je taj element jednak nekom broju s_2 a zatim proveravamo da li je taj broj jednak indeksu ***i*** ili nekom od ranije upisanih brojeva u tabeli. U zavisnosti od ove provere razlikujemo sledeće mogućnosti:

K 4.2.1: Ako je onda se s_2 upisuje u tabelu ispred broja i dalje se posupa kao u slučaju opisanom u kopraku K4.1.

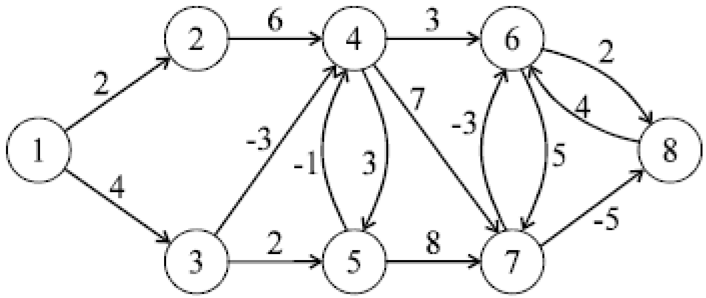
K 4.2.2: Ako je s_2 jednak bilo kome od ranije upisanih brojeva u tabeli, onda se u tabelu ispred broja s_1 upisuje broj jednak indeksu $\boldsymbol{i}$, a dalje identifikacija najkraćeg puta je ista kao u koraku K4.1.

K 4.2.3: Ako je broj s_2 različit i od indeksa $\boldsymbol{i}$ i od svih brojeva ranije upisanih u tabelu, onda uočimo element $d_{1s_2}^{n}$ matrice D^n. Neka je taj element jednak broju s_3. Proverimo da li je broj s_3 jednak indeksu $\boldsymbol{i}$ ili nekom od ranije upisanih brojeva i u zavisnosti od te provere koristi se neka od mogućnosti opisana u koracima K 4.2.1, K4.2.2 i K 4.2.3.

Korak 4.2.3. se ponavlja se dok ne nastupi situacija iz koraka 4.2.1 ili 4.2.2 kada se posupak identifikacije najkraćeg puta između čvorova $\boldsymbol{i}$ i $\boldsymbol{j}$ završava a brojevi upisani u tabelu predstavljaju oznake (idekse) čvorova preko kojih ide najkraći put p_{ij} sa dužinom $d(p_{ij}) = c_{ij}^{n}$. Ovaj postupak identifikacije najkraćih puteva može se izvesti za svako $i \in N$ i svako $j \in N$ i na taj način mogu se identifikovati najkraći putevi između svaka dva čvora ***i*** i ***j*** date mreže.

PRIMER 88 (preuzeto iz literature sa FONa)

Za datu mrežu naći dužine najkraćih puteva između svaka dva čvora



Rešenje:

$k = 0$

$c_{ij}^0 = c_{ij} \quad \forall (i,j) \in L$

$c_{ii}^0 = 0 \; \forall i \in N$

$c_{ij}^0 = \infty \;\; \forall (i,j) \notin L$

$$C^0 = C = \begin{array}{cc} & I \\ & \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ J \; \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} & \begin{pmatrix} 0 & 2 & 4 & \infty & \infty & \infty & \infty & \infty \\ \infty & 0 & \infty & 6 & \infty & \infty & \infty & \infty \\ \infty & \infty & 0 & -3 & 2 & \infty & \infty & \infty \\ \infty & \infty & \infty & 0 & 3 & 3 & 7 & \infty \\ \infty & \infty & \infty & -1 & 0 & \infty & 8 & \infty \\ \infty & \infty & \infty & \infty & \infty & 0 & 5 & 2 \\ \infty & \infty & \infty & \infty & \infty & -3 & 0 & -5 \\ \infty & \infty & \infty & \infty & \infty & 4 & \infty & 0 \end{pmatrix} \end{array}$$

$$D^0 = D = \begin{array}{cc} & I \\ & \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ J \; \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} & \begin{pmatrix} \infty & 2 & 3 & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 4 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 4 & 5 & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & 5 & 6 & 7 & \infty \\ \infty & \infty & \infty & 4 & \infty & \infty & 7 & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & 7 & 8 \\ \infty & \infty & \infty & \infty & \infty & 6 & \infty & 8 \\ \infty & \infty & \infty & \infty & \infty & 6 & \infty & \infty \end{pmatrix} \end{array}$$

$k = 1$

$I = \emptyset$ elementi k-te kolone različiti od ∞, a da se ne nalaze na dijagonali.

$J = \{2,3\}$ elementi k-tog reda različiti od ∞ a da se ne nalaze na dijagonali.

Pošto je $I = \emptyset$ matrica C^1 će ostati ista kao matrica C^0, $D^1 = D^0$.

$k = 2$

$I = \{1\}, J = \{4\}$ razmatramo samo element $c_{14}^2 = \min\{c_{14}^1, c_{12}^1 + c_{24}^1\} = \min\{\infty, 2 + 6\} = 8$

Matrica C^2 se razlikuje od matrica C^1 i C^0 samo u elementu c_{14}^2 koji je ranije imao vrednost ∞ a sada 8.

Matrica D^2 se takođe menja: $d_{14}^2 = 2$

$k = 3$

$I = \{1\}, J = \{4,5\}$ razmatramo

$$c_{14}^{3} = \min\{c_{14}^{2}, c_{13}^{2} + c_{34}^{2}\} = \min\{8, 4 - 3\} = 1$$
$$c_{15}^{3} = \min\{c_{15}^{2}, c_{13}^{2} + c_{35}^{2}\} = \min\{\infty, 4 + 2\} = 6$$

Matrica C^3 se sada razlikuje od prethodnih matrica u elementima c_{14}^{3} i c_{15}^{3}

Matrica D^3 se sada razlikuje od prethodnih matrica u elementima $d_{14}^{3} = 3$ i $d_{15}^{3} = 3$.

$$C^3 = \begin{array}{ccc} & & I \\ & & \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ J & \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} & \begin{pmatrix} 0 & 2 & 4 & 1 & 6 & \infty & \infty & \infty \\ \infty & 0 & \infty & 6 & \infty & \infty & \infty & \infty \\ \infty & \infty & 0 & -3 & 2 & \infty & \infty & \infty \\ \infty & \infty & \infty & 0 & 3 & 3 & 7 & \infty \\ \infty & \infty & \infty & -1 & 0 & \infty & 8 & \infty \\ \infty & \infty & \infty & \infty & \infty & 0 & 5 & 2 \\ \infty & \infty & \infty & \infty & \infty & -3 & 0 & -5 \\ \infty & \infty & \infty & \infty & \infty & 4 & \infty & 0 \end{pmatrix} \end{array}$$

$k = 4$

$I = \{1,2,3,5\}$, $J = \{5,6,7\}$ razmatramo

$$c_{15}^{4} = \min\{c_{15}^{3}, c_{14}^{3} + c_{45}^{3}\} = \min\{6, 1 + 3\} = 4$$
$$c_{16}^{4} = \min\{c_{16}^{3}, c_{14}^{3} + c_{46}^{3}\} = \min\{\infty, 1 + 3\} = 4$$
$$\dots$$
$$c_{36}^{4} = \min\{c_{36}^{3}, c_{34}^{3} + c_{46}^{3}\} = \min\{\infty, -3 + 3\} = 0$$
$$c_{37}^{4} = \min\{c_{37}^{3}, c_{34}^{3} + c_{47}^{3}\} = \min\{\infty, -3 + 7\} = 4$$
$$c_{55}^{4} = \min\{c_{55}^{3}, c_{54}^{3} + c_{45}^{3}\} = \min\{0, -1 + 3\} = 0$$
$$c_{56}^{4} = \min\{c_{56}^{3}, c_{54}^{3} + c_{46}^{3}\} = \min\{\infty, -1 + 3\} = 2$$
$$c_{57}^{4} = \min\{c_{57}^{3}, c_{54}^{3} + c_{47}^{3}\} = \min\{8, -1 + 7\} = 6$$

$$C^4 = \begin{array}{ccc} & & I \\ & & \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ J & \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} & \begin{pmatrix} 0 & 2 & 4 & 1 & 4 & 4 & 8 & \infty \\ \infty & 0 & \infty & 6 & 9 & 9 & 13 & \infty \\ \infty & \infty & 0 & -3 & 0 & 0 & 4 & \infty \\ \infty & \infty & \infty & 0 & 3 & 3 & 7 & \infty \\ \infty & \infty & \infty & -1 & 0 & 2 & 6 & \infty \\ \infty & \infty & \infty & \infty & \infty & 0 & 5 & 2 \\ \infty & \infty & \infty & \infty & \infty & -3 & 0 & -5 \\ \infty & \infty & \infty & \infty & \infty & 4 & \infty & 0 \end{pmatrix} \end{array}$$

$$D^4 = \begin{array}{ccc} & & I \\ & & \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ J & \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} & \begin{pmatrix} \infty & 2 & 3 & 3 & 4 & 4 & \infty & \infty \\ \infty & \infty & \infty & 4 & 4 & 4 & 4 & \infty \\ \infty & \infty & \infty & 4 & 5 & 4 & 4 & \infty \\ \infty & \infty & \infty & \infty & 5 & 6 & 7 & \infty \\ \infty & \infty & \infty & 4 & 4 & 4 & 4 & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & 7 & 8 \\ \infty & \infty & \infty & \infty & \infty & 6 & \infty & 8 \\ \infty & \infty & \infty & \infty & \infty & 6 & \infty & \infty \end{pmatrix} \end{array}$$

Zbog opširnosti preskačemo nekoliko iteracija.

Za $k = 7$ se dobija sledeća matrica rastojanja:

$$C^7 = \begin{array}{ccc} & & I \\ & & \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ J & \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} & \begin{pmatrix} 0 & 2 & 4 & 1 & 4 & 4 & 8 & 3 \\ \infty & 0 & \infty & 6 & 9 & 9 & 13 & 8 \\ \infty & \infty & 0 & -3 & 0 & 0 & 4 & -1 \\ \infty & \infty & \infty & 0 & 3 & 3 & 7 & 2 \\ \infty & \infty & \infty & -1 & 0 & 2 & 6 & 1 \\ \infty & \infty & \infty & \infty & \infty & 0 & 5 & 0 \\ \infty & \infty & \infty & \infty & \infty & -3 & 0 & -5 \\ \infty & \infty & \infty & \infty & \infty & 4 & 9 & 0 \end{pmatrix} \end{array}$$

Dalje, imamo za $k = 8$

$I = \{1,2,3,4,5,6,7\}, J = \{6,7\}$.

Za sledeće elemente matrice C^8 izračunavamo nove vrednosti:

$$C_{16}^8 = 4, C_{17}^8 = 8,$$
$$C_{26}^8 = 9, C_{27}^8 = 13,$$
$$C_{36}^8 = 0, C_{37}^8 = 4,$$
$$C_{46}^8 = 3, C_{47}^8 = 7$$
$$, C_{56}^8 = 2, C_{57}^8 = 6,$$
$$C_{66}^8 = 0, C_{67}^8 - 5,$$
$$C_{76}^8 = -3, C_{77}^8 = 0.$$

Primetimo da je matrica $C^8 = C^7$.

$$D^8 = \begin{array}{ccc} & & I \\ & & \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ J & \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} & \begin{pmatrix} \infty & 2 & 3 & 3 & 4 & 4 & 4 & 7 \\ \infty & \infty & \infty & 4 & 4 & 4 & 4 & 7 \\ \infty & \infty & \infty & 4 & 5 & 4 & 4 & 7 \\ \infty & \infty & \infty & 5 & 4 & 6 & 7 & 7 \\ \infty & \infty & \infty & 5 & 4 & 4 & 4 & 7 \\ \infty & \infty & \infty & \infty & \infty & 7 & 6 & 6 \\ \infty & \infty & \infty & \infty & \infty & 6 & 6 & 7 \\ \infty & \infty & \infty & \infty & \infty & 8 & 6 & 7 \end{pmatrix} \end{array}$$

Pošto je $k = n$ i svi $c_{ii}^8 = 0, i = 1, .., 8$ sledi da je algoritam došao do kraja.

Poslednja matrica predstavlja rešenje zadataka. Ostalo je da rekonstruištemo putanju (na času formirati matricu D i rekonstrukciju najkraćeg puta).

Putanja se rekonstruiše pomoću matrice D:

Put 1-1 ne postoji ($d_{11}^8 = \infty$)

Put 1-2 ide preko čvora 2 ($d_{12}^8 = 2$) 1-2

Put 1-3 ide preko čvora 3 ($d_{13}^8 = 3$) 1-3

Put 1-4 ide preko čvora 3 ($d_{14}^8 = 3$) 1-3-4

Put 1-5 ide preko čvora 4 ($d_{15}^8 = 4$) 1-3-4-5

Put 1-6 ide preko čvora 4 ($d_{16}^8 = 4$) 1-3-4-6

Put 1-7 ide preko čvora 4 ($d_{17}^8 = 4$) 1-3-4-7

Put 1-8 ide preko čvora 7 ($d_{18}^8 = 7$) 1-3-4-7-8

Put 2-1 ne postoji ($d_{21}^8 = \infty$)

Put 2-2 ne postoji ($d_{22}^8 = \infty$)

Put 2-3 ne postoji ($d_{23}^8 = \infty$)

Put 2-4 ide preko čvora 4 $(d_{24}^{8} = 4)$ 2-4
Put 2-5 ide preko čvora 4 $(d_{25}^{8} = 4)$ 2-4-5
Put 2-6 ide preko čvora 4 $(d_{26}^{8} = 6)$ 2-4-6
Put 2-7 ide preko čvora 4 $(d_{27}^{8} = 4)$ 2-4-7
Put 2-8 ide preko čvora 7 $(d_{28}^{8} = 6)$ 2-4-6-8

Put 3-1 ne postoji $(d_{31}^{8} = \infty)$
Put 3-2 ne postoji $(d_{32}^{8} = \infty)$
Put 3-3 ne postoji $(d_{33}^{8} = \infty)$
Put 3-4 ide preko čvora 4 $(d_{34}^{8} = 4)$ 3-4
Put 3-5 ide preko čvora 4 $(d_{35}^{8} = 5)$ 3-4-5
Put 3-6 ide preko čvora 4 $(d_{36}^{8} = 4)$ 3-4-6
Put 3-7 ide preko čvora 4 $(d_{37}^{8} = 4)$ 3-4-7
Put 3-8 ide preko čvora 7 $(d_{38}^{8} = 7)$ 3-4-7-8

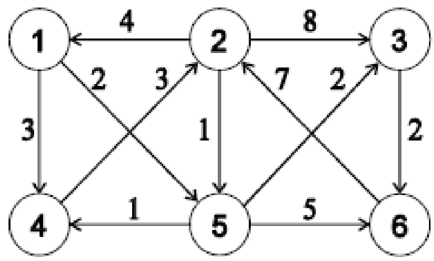
Put 4-1 ne postoji $(d_{41}^{8} = \infty)$
Put 4-2 ne postoji $(d_{42}^{8} = \infty)$
Put 4-3 ne postoji $(d_{43}^{8} = \infty)$
Put 4-4 ide preko čvora 5 $(d_{44}^{8} = 5)$ 4-5-4
Put 4-5 ide preko čvora 4 $(d_{45}^{8} = 4)$ 4-5
Put 4-6 ide preko čvora 6 $(d_{46}^{8} = 6)$ 4-6
Put 4-7 ide preko čvora 7 $(d_{47}^{8} = 7)$ 4-7
Put 4-8 ide preko čvora 6 $(d_{48}^{8} = 7)$ 4-7-8

Put 5-1 ne postoji $(d_{51}^{8} = \infty)$
Put 5-2 ne postoji $(d_{52}^{8} = \infty)$
Put 5-3 ne postoji $(d_{53}^{8} = \infty)$
Put 5-4 ide preko čvora 5 $(d_{54}^{8} = 5)$ 5-4
Put 5-5 ide preko čvora 4 $(d_{55}^{8} = 4)$ 5-4-5
Put 5-6 ide preko čvora 6 $(d_{56}^{8} = 4)$ 5-4-6
Put 5-7 ide preko čvora 7 $(d_{57}^{8} = 4)$ 5-4-7
Put 5-8 ide preko čvora 6 $(d_{58}^{8} = 7)$ 5-4-7-8

Put 6-1 ne postoji $(d_{61}^{8} = \infty)$
Put 6-2 ne postoji $(d_{62}^{8} = \infty)$
Put 6-3 ne postoji $(d_{63}^{8} = \infty)$
Put 6-4 ne postoji $(d_{64}^{8} = \infty)$
Put 6-5 ne postoji $(d_{65}^{8} = \infty)$
Put 6-6 ide preko čvora 7 $(d_{66}^{8} = 7)$ 6-7-6
Put 6-7 ide preko čvora 6 $(d_{67}^{8} = 6)$ 6-7
Put 6-8 ide preko čvora 6 $(d_{68}^{8} = 6)$ 6-8
i tako dalje.

PRIMER 89 (preuzeto iz literature sa FONa)

Data je mreža:



Slika 1.13.

Naći najkraće puteve između svaka dva čvora u mreži.

Rešenje:

$$C^0 = C = \begin{bmatrix} 0 & \infty & \infty & 3 & 2 & \infty \\ 4 & 0 & 8 & \infty & 1 & \infty \\ \infty & \infty & 0 & \infty & \infty & 2 \\ \infty & 3 & \infty & 0 & \infty & \infty \\ \infty & \infty & 2 & 1 & 0 & 5 \\ \infty & 7 & \infty & \infty & \infty & 0 \end{bmatrix}$$

$k = 1$, $I = \{2\}$, $J = \{4, 5\}$
$c_{24} = \min\{\infty, 4+3\} = 7$ *
$c_{25} = \min\{1, 4+2\} = 1$

U matrici je promenjena vrednost jedino elementa c_{24} (označen zvezdicom) i to sa ∞ na 7. Svi ostali elementi matrice C^1 su isti kao elementi matrice C^0.

$k = 2, I = \{4, 6\}, J = \{1, 3, 4, 5\}$
$c_{41} = \min\{\infty, 3+4\} = 7$ *
$c_{43} = \min\{\infty, 3+8\} = 11$ *
$c_{44} = \min\{0, 3+7\} = 0$
$c_{45} = \min\{\infty, 3+1\} = 4$ *
$c_{61} = \min\{\infty, 7+4\} = 11$ *
$c_{63} = \min\{\infty, 7+8\} = 15$ *
$c_{64} = \min\{\infty, 7+7\} = 14$ *
$c_{65} = \min\{\infty, 7+1\} = 8$ *

Matrica C^2 ima oblik:

$$C^2 = \begin{bmatrix} 0 & \infty & \infty & 3 & 2 & \infty \\ 4 & 0 & 8 & 7 & 1 & \infty \\ \infty & \infty & 0 & \infty & \infty & 2 \\ 7 & 3 & 11 & 0 & 4 & \infty \\ \infty & \infty & 2 & 1 & 0 & 5 \\ 11 & 7 & 15 & 14 & 8 & 0 \end{bmatrix}$$

U sledećim koracima biće date samo matrice C^k, $k = 3, \ldots$

$$C^3 = \begin{bmatrix} 0 & \infty & \infty & 3 & 2 & \infty \\ 4 & 0 & 8 & 7 & 1 & 10 \\ \infty & \infty & 0 & \infty & \infty & 2 \\ 7 & 3 & 11 & 0 & 4 & 13 \\ \infty & \infty & 2 & 1 & 0 & 4 \\ 11 & 7 & 15 & 14 & 8 & 0 \end{bmatrix} \qquad C^4 = \begin{bmatrix} 0 & 6 & 14 & 3 & 2 & 16 \\ 4 & 0 & 8 & 7 & 1 & 10 \\ \infty & \infty & 0 & \infty & \infty & 2 \\ 7 & 3 & 11 & 0 & 4 & 13 \\ 8 & 4 & 2 & 1 & 0 & 4 \\ 11 & 7 & 15 & 14 & 8 & 0 \end{bmatrix}$$

$$C^5 = \begin{bmatrix} 0 & 6 & 4 & 3 & 2 & 6 \\ 4 & 0 & 3 & 2 & 1 & 5 \\ \infty & \infty & 0 & \infty & \infty & 2 \\ 7 & 3 & 6 & 0 & 4 & 8 \\ 8 & 4 & 2 & 1 & 0 & 4 \\ 11 & 7 & 10 & 9 & 8 & 0 \end{bmatrix} \qquad C^6 = \begin{bmatrix} 0 & 6 & 4 & 3 & 2 & 6 \\ 4 & 0 & 3 & 2 & 1 & 5 \\ 13 & 9 & 0 & 11 & 10 & 2 \\ 7 & 3 & 6 & 0 & 4 & 8 \\ 8 & 4 & 2 & 1 & 0 & 4 \\ 11 & 7 & 10 & 9 & 8 & 0 \end{bmatrix}$$

Poslednja matrica je rešenje zadatka.
Rekonstrukcija rešenja.