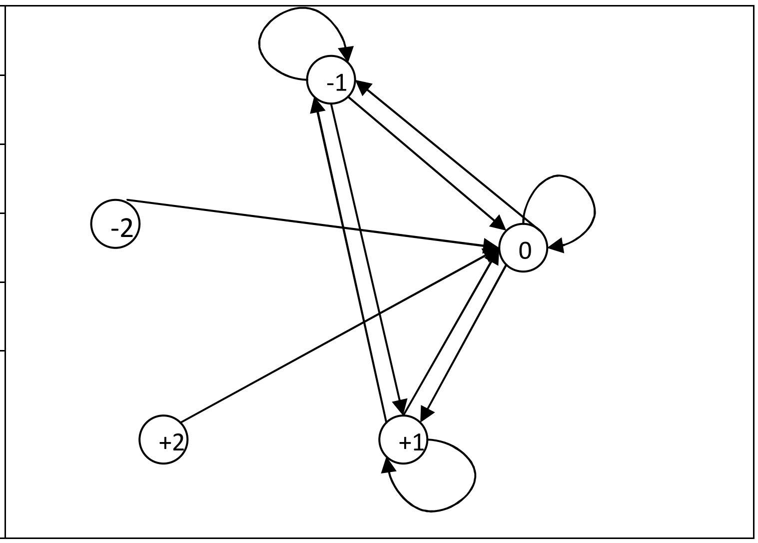
Grafovi

1. Uvod

U okviru predmeta Informatika za bibliotekare govorili smo o relacijama koje su definisane na nekom skupu: relacija je definisana na nekom skupu A ako za svaka dva elementa skupa A možemo da kažemo da jesu u relaciji ili da nisu u relaciji. Relacija se na nekom skupu može predstaviti tablicom ali i usmerenim grafom. Pogledajmo primer.

Na skupu {-2, -1, 0, +1, +2} definisana je relacija R na sledeći način: xRy ako i samo ako je $x^2 + 2y^2 \leq 5$. Predstaviti relaciju R tablicom, usmerenim grafom i ispitati svojstva relacije R.

	-2	**-1**	**0**	**+1**	**+2**
-2	F	F	T	F	F
-1	F	T	T	T	F
0	F	T	T	T	F
+1	F	T	T	T	F
+2	F	F	T	F	F



Kakve su osobine ove relacije?

1. *Refleksivnost*: Relacija je refleksivna ako $\forall x \in A$ važi da je xRx. Kako se vidi da li je relacija refleksivna iz tablice? Sve vrednosti na dijagonali treba da budu T (true). Kako se vidi da li je relacija refleksivna iz usmerenog grafa? Svaki čvor u grafu treba da ima luk koji pokazuje samog sebe. Naša relacija nije refleksivna jer svi čvorovi nemaju luk koji pokazuje samog sebe (-2 i +2). Relacija nije ni irefleksivna jer neki čvorovi imaju luk ka samom sebi (-1, 0, +1).

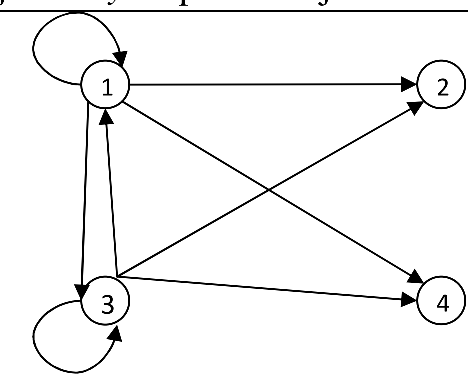
2. *Simetričnost*: relacija je simetrična ako $\forall x, y \in A$ koji ne moraju biti različiti važi da ako je $xRy \Rightarrow yRx$. Kako se vidi da li je relacija simetrična iz tablice? Sve vrednosti u tablici moraju da budu simetrične (iste) u odnosu na dijagonalu. Kako se vidi da li je relacija simetrična iz usmerenog grafa? Ako postoji luk iz čvora x u čvor y onda mora postojati i luk iz čvora y u čvor x. Relacija R nije simetrična jer

to ne važi za svaka dva čvora; postoji luk iz čvora -2 u čvor 0, ali ne postoji luk iz čvora 0 u čvor -2. Relacija nije ni asimetrična jer za neka dva čvora važi da postoje lukovi u oba smera (npr. čvorovi -1 i +1).

3. *Tranzitivnost*: relacija je tranzitivna ako $\forall x, y, z \in A$ koji ne moraju biti različiti važi da ako je xRy i $yRz \Rightarrow xRz$. Kako se vidi da li je relacija tranzitivna iz tablice? To se iz tablice može teško uočiti. Kako se vidi da li je relacija tranzitivna iz usmerenog grafa? Ako postoji luk iz čvora x ka čvoru y i postoji luk iz čvora y ka čvoru z, onda mora postojati i luk iz čvora x ka čvoru z. Naša relacija nije tranzitivna jer postoji luk iz čvora +2 ka čvoru 0 i postoji luk iz čvora 0 ka čvoru +1, a ne postojati luk iz čvora +2 ka čvoru +1. Relacija nije ni itranzitivna jer postoji luk iz čvora -1 ka čvoru 0 i postoji luk iz čvora 0 ka čvoru +1 i postojati luk iz čvora -1 ka čvoru +1.

Zadatak 1: Neka je relacija R definisana na skupu $\{1, 2, 3, 4\}$ na sledeći način: xRy ako i samo ako je $x + 2y$ neparan broj.

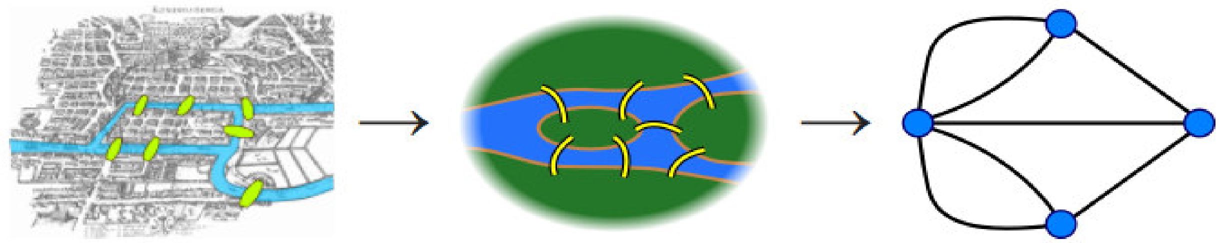
	1	**2**	**3**	**4**
1	T	T	T	T
2	F	F	F	F
3	T	T	T	T
4	F	F	F	F



Relacija je nerefleksivna, nesimetrična i tranzitivna. Zašto? Tranzitivnost: Dva broja su u relaciji R samo ako je prvi broj neparan. Dakle ako je xRy, to znači da je x neparno pa će biti u relaciji sa bilo kojim drugim brojem iz skupa.

☺

2. Neusmereni grafovi

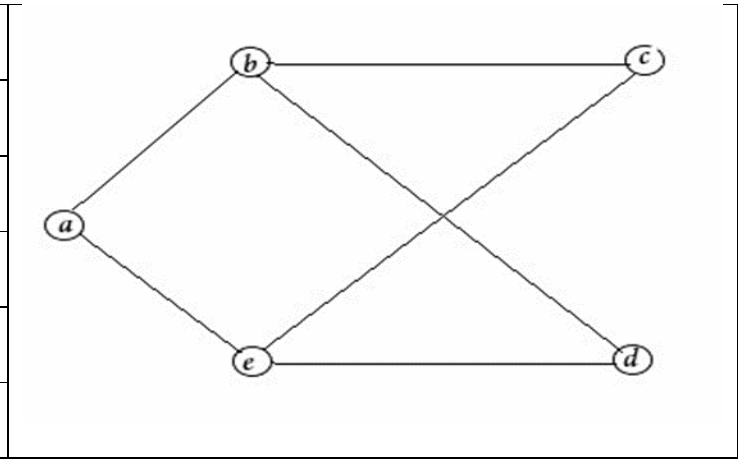


Problem sedam mostova grada Kenigsberga (sada je to Kalinjingrad u Rusiji) preko reke Pregel je bio ovako postavljen: može li čovek krećući s nekog mesta da pređe sve mostove tačno jedanput i da se vrati u polaznu tačku. Čuveni švajcarski matematičar Leonard Ojler (Leonhard Euler) je dao negativan odgovor na to pitanje. Njegovo bavljenje ovim problemom postavilo je osnove matematičke discipline teorije grafova. On je zaključio da kako se čovek kreće po povezanom delu zemlje nije ni od kakvog značaja, važno je samo kojim redom se prelaze mostovi. Ceo problem je posmatrao apstraktno: zadržao je samo povezane površine zemlje (to su čvorovi) i mostove koji ih povezuju (to su lukovi). Tako se dobija apstraktna struktura koja se naziva *graf*. Ojler je uočio da kad god se nekim lukom stigne u jedan čvor, on se mora napustiti nekim drugim lukom, i to se ponavlja koliko god puta se prolazilo tim čvorom (ali uvek je potreban drugi par lukova). To znači, da bi bio moguć *Ojelerov put* – polazak iz jednog čvora i povratak u njega koristeći sve lukove tačno jedanput – potrebno je da svi čvorovi imaju paran broj lukova. Kasnije je dokazano da važi i obrnuto: ako se u nekom grafu u svakom čvoru sreće paran broj lukova, u takvom grafu postoji Ojlerov put. Graf koji apstrahuje problem sedam mostova grada Kenigsberga je *neusmereni graf*, jer se svaki most, naravno, može preći u oba smera.

Graf se definiše kao dvojka $G = (A, L)$, gde je A skup čvorova grafa, a L skup lukova koji spajaju čvorove. Zapravo je L skup koji sadrži (neuređene) dvojke (a_1, a_2), pri čemu $a_1, a_2 \in A$. Graf G je *jednostavan* ako nijedan čvor ne sadrži petlju, tj. skup L ne sadrži dvojku (a, a), i ako ne postoji dvostruki luk između dva čvora. Graf koji predstavljava mostove grada Kenigsberga nije jednostavan graf iako ne sadrži petlje (nijedan most ne povezuje isto parče zemlje sa samim sobom!), ali postoje dvostruki lukovi (postoji više mostova koji spajaju dve iste površine zemlje).

Zadatak 2. Nacrtati jednostavan neusmeren graf $G = (A, L)$ čiji je skup čvorova $A = \{a, b, c, d, e\}$ i skup lukova $L = \{ab, ae, bc, bd, ce, de\}$ i predstaviti ga u obliku tablice.

	a	b	c	d	e
a	F	T	F	F	T
b	T	F	T	T	F
c	F	T	F	F	T
d	F	T	F	F	T
e	T	F	T	T	F



Treba uočiti da je tablica neusmerenog grafa uvek simetrična u odnosu na dijagonalu, jer ako je čvor a povezan lukom sa čvorom b, onda je i čvor b povezan lukom s čvorom a.

Put dužine k u jednom grafu je niz različitih čvorova a_0, a_1, ..., a_k takvih da su $a_{i-1}\ a_i$ lukovi grafa za $i = 1, ..., k$. Jedan takav put se označava sa $a_0a_1...a_k$. Na primer, u gornjem zadatku je $a\ b\ d\ e$ jedan put dužine 3.

Ciklus u grafu je niz čvorova a_0, a_1, ..., a_k takvih da su $a_{i-1}\ a_i$ lukovi grafa za $i = 1, ..., k$ i da je $a_0 = a_k$, pri čemu se nijedan drugi čvor ne ponavlja. Na primer, u gornjem zadatku je $a\ b\ d\ e\ a$ jedan ciklus.

Aciklični grafovi su grafovi koji ne sadrže nijedan ciklus. *Drveta* su primer acikličnog grafa.

Jedan graf je *povezan* ako postoji put između svaka dva čvora. U principu se svaki graf može razdeliti u podgrafove od kojih je svaki povezan. *Podgraf* jednog grafa $G = (A, L)$ je neki graf $G' = (A', L')$ takav da je $A' \subseteq A$ i $L' \subseteq L$. Najmanji broj takvih povezanih grafova na koji se neki graf može raspodeliti se naziva *povezanost grafa* i obeležava se sa $c(G)$. Povezanost grafa je pojam koji je veoma važan u konstrukciji računarskih mreža.

Algoritam za izračunavanje povezanosti grafa – Neka je $G = (A, L)$ neki graf.
početak
$A' = A$ /* na početku su svi čvorovi u A' */
$c = 0$ /* broj povezanih podgrafova je na početku 0 */
sve dok je $A' \neq \emptyset$ **ponavljaj**
početak
izaberi $y \in A'$
nađi sve čvorove koji su sa y povezani nekim putem
izbaci sve te čvorove kao i y iz A' a odgovarajuće lukove iz L

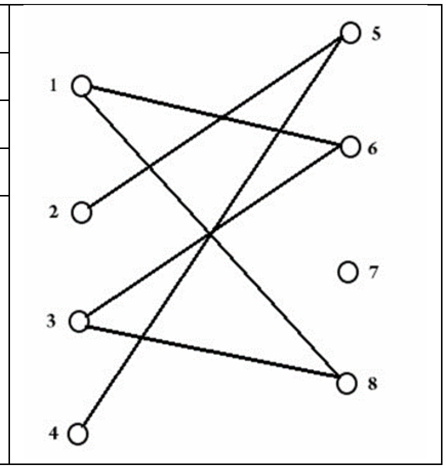
$c = c + 1$

kraj

kraj

Rad algoritma je prikazan za graf na slici sledećom tabelom (različitom bojom su prikazani čvorovi koji se izbacuju iz skupa A' u svakom prolasku kroz petlju). Pokazuje se da ovaj graf ima povezanost 3, $c(G) = 3$, jer je najmanji broj povezanih grafova u koji se može razdeliti 3: $G_1 = \{\{1, 3, 6, 8\}, \{1\text{-}6, 1\text{-}8, 3\text{-}6, 3\text{-}8\}\}$, $G_2 = \{\{2, 4, 5\}, \{2\text{-}5, 4\text{-}5\}\}$ i $G_3 = \{\{7\}, \varnothing\}$.

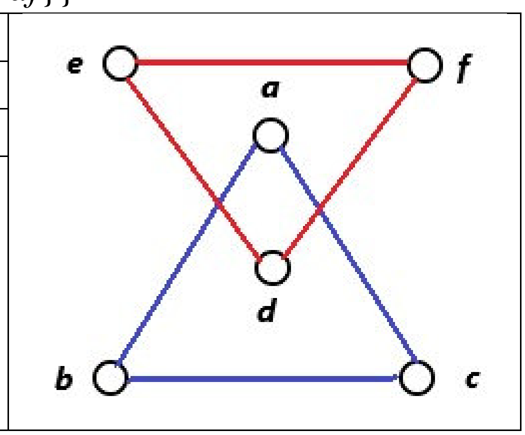
	A'	c
početni čvorovi	$\{1, 2, 3, 4, 5, 6, 7, 8\}$	0
kad se izabere $y = 1$	$\{2, 4, 5, 7\}$	1
kad se izabere $y = 2$	$\{7\}$	2
kad se izabere $y = 7$	$\varnothing$	3



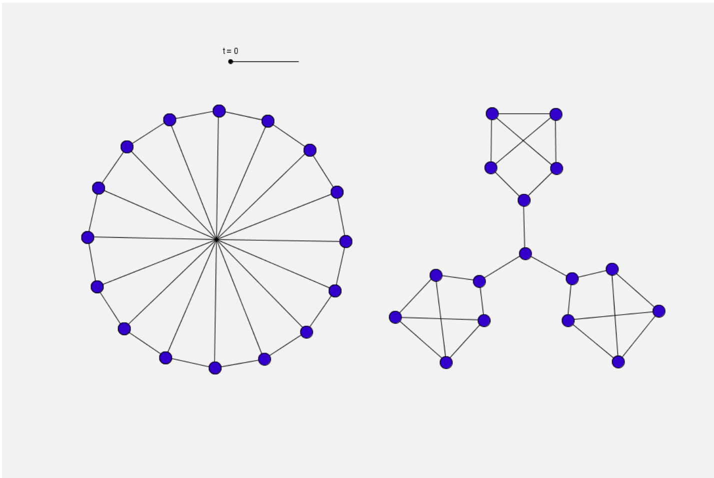
Zadatak 3. Predstaviti graf sa slike u obliku $G = (A, L)$. Prikazati rad algoritma za izračunavanje povezanosti grafa za graf na slici i predstaviti njegove povezane oblike.

$G = \{\{a, b, c, d, e, f\}, \{ab, bc, ca, ed, fe, df\}\}$

	A'	c
početni čvorovi	$\{a, b, c, d, e, f\}$	0
kad se izabere $y = a$	$\{ d, e, f \}$	1
kad se izabere $y = d$	$\varnothing$	2

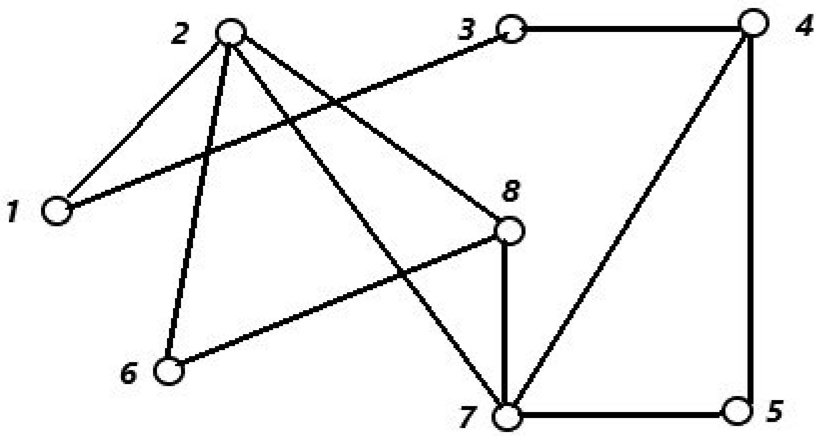


$C(G) = 2$; $G_1 = \{\{a, b, c\}, \{ ab, bc, ca \}\}$, $G_2 = \{\{ d, e, f \}, \{ ed, fe, df \}\}$ ☺



Problem sličan pronalaženju Ojlerovog puta je problem pronalaženja ciklusa u grafu takvih da se svakim čvorom prođe samo jednom. Takav ciklus se zove *Hamiltonov ciklus*, a takvi grafovi *Hamiltonovi grafovi*.[1] Ovi grafovi se javljuju u rešavanju problema redova vožnje, telekomunikacionih mreža, i drugih. Za razliku od problema Ojlerovih ciklusa, problem nalaženja Hamiltonovih ciklusa je vrlo težak. Da bi se proverilo da li je graf Ojlerov, dovoljno je znati stepene njegovih čvorova. Za utvrđivanje da li je graf Hamiltonov, to nije dovoljno. Ako se pogledaju dva grafa prikazana na slici, oni oba imaju po 16 čvorova stepena 3 (u svaki čvor ulaze(izlaze 3 luka), ali je prvi Hamiltonov, a drugi očigledno nije.[2] Za razliku od Ojlerovih grafova, ne postoji jednostavno pravilo kojim se određuje da li u grafu postoji Hamiltonov ciklus. To je jedan od najvažnijih problema u teoriji grafova koji još uvek nije rešen.

Zadatak 4. Pronaći najduži Hamiltonov ciklus u grafu sa slike.



[1] Ovo je poznati problem koji je ime dobio po irskom matematičaru Vilijemu Hamiltonu, koji je predložio popularnu igru zasnovanu na ovom problemu 1857. godine.

[2] Slika je preuzeta sa stranice http://alas.matf.bg.ac.rs/~ml07199/algoritmi-final/Hamiltonovi_ciklusi.html.

Najduži ciklus prolazi kroz sve čvorove ovim redom (ako se krene od čvora 1): 1, 2, 6, 8, 7, 5, 4, 3, 1. Još neki cilusi dužine 7: bi bili 1 2 6 8 7 4 3 1 i 1 3 4 5 7 8 2 1.

☺

Hamiltonovi grafovi omogućavaju da se modeliraju mnogi konkretni problemi, među kojima je i klasičan problem *trgovačkog putnika*. Taj problem bi se mogao ovako formulisati: Trgovački putnik treba da obiđe određen broj gradova koji su povezani putevima. Treba pronaći put koji prolazi kroz sve gradove samo jednom, minimizirajući pri tom cenu putovanja.

U odgovarajućem modelu Hamiltonovog grafa čvorovi su gradovi, lukovi putevi, pri čemu se svakom putu pridružuje *težina* koja predstavlja cenu putovanja (npr. dužina puta, trajanje putovanja i sl.) Da bi se rešio ovaj problem, treba pronaći Hamiltonov put čija je težina najmanja. Kao što smo rekli, ne postoji efikasan algoritam za pronalaženje optimalnog puta, ali postoji efikasan algoritam za pronalaženje skoro optimalnog rešenja (rešenja koje je bolje od većine drugih).

Algoritam najbližeg suseda - Ovaj algoritam pronalazi skoro optimalno rešenje za problem trgovačkog putnika tako što generiše Hamiltonov ciklus za težinski graf koji sadrži skup čvorova A. Rešenje je na kraju pronađeni put i njegova ukupna težina w.

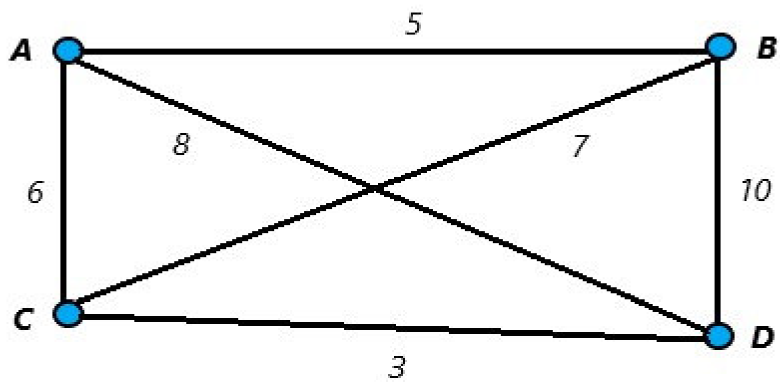
početak

    izaberi $a \in A$ /* mesto odakle trgovački putnik kreće */
    $put = a$
    $w = 0$
    $a' = a$
    označi a' /* kroz označene čvorove se prošlo i u njih se više ne ide */
    sve dok postoje čvorovi koji nisu označeni **ponavljaj**
    početak
        Izaberi neki neoznačeni čvor b koji je najbliži a'
        $put = put\ b$
        $w = w + težina\_luka(a'b)$
        $a' = b$
        označi a'
    kraj
    /* prošlo se kroz sve čvorove */
    $put = put\ a$ /* povratak na početak */

$w = w + težina\_luka(a'a)$

kraj

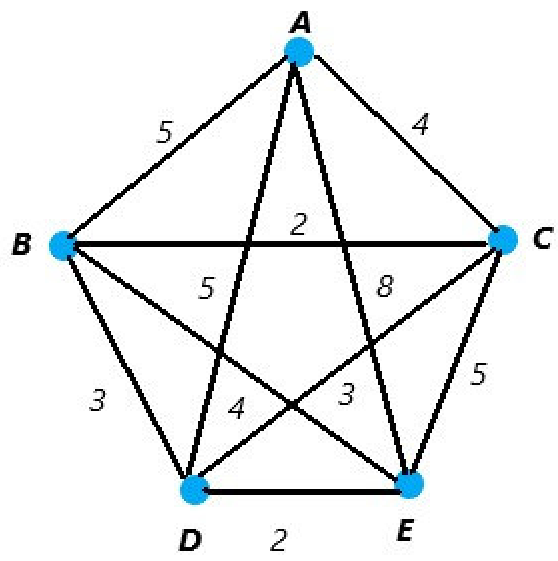
Rad algoritma je za graf na slici prikazan tabelom, ako se polazi iz čvora *D*.



	b	put	a	w	a'	skup označenih čvorova
na početku	-	*D*	*D*	*0*	*D*	*D*
petlja	*C*	*DC*	*(D)*	3	*C*	*D C*
	A	*DCA*	*(D)*	9	*A*	*D C A*
	B	*DCAB*	*(D)*	14	*B*	*D C A B*
na kraju	*B*	*DCABD*	*(D)*	24	(*B*)	*(D C A B)*

Rezultat rada algoritma je Hamiltonov ciklus *DCBAD* čija je ukupna težina 24. Kada bi se iscrpno prošlo kroz sve mogućnosti našla bi se još dva Hamiltonova ciklusa: *ABCDA* težine 23 i *ACBDA* težine 31. U kompletnom grafu koji ima 20 čvorova (svaka dva čvora su povezana lukom) bilo bi oko $6{,}1 \times 10^{16}$ Hamiltonovih ciklusa. Ispitivanje svakog od njih bi zahtevalo jako puno računarskog vremena.

Zadatak 5. Primeniti algoritam najbližeg suseda na graf sa slike za pronalaženje Hamiltonovog ciklusa polazeći od čvora *A*. Rad algoritma prikazati tabelom.



	b	*put*	*a*	*w*	*a'*	*skup označenih čvorova*
na početku	-	*A*	*A*	0	*A*	*A*
petlja	*C*	*AC*	*(A)*	4	*C*	*A C*
	B	*ACB*	*(A)*	6	*B*	*A C B*
	D	*ACBD*	*(A)*	9	*D*	*A C B D*
	E	*ACBDE*	*(A)*	11	*E*	*A C B D E*
na kraju	*E*	*ACBDEA*	*(A)*	19	(*E*)	*(A C B D E)*

Kako bi izgledao Hamiltonov ciklus ako bi put krenuo iz čvora *D* i kolika bi bila njegova težina?

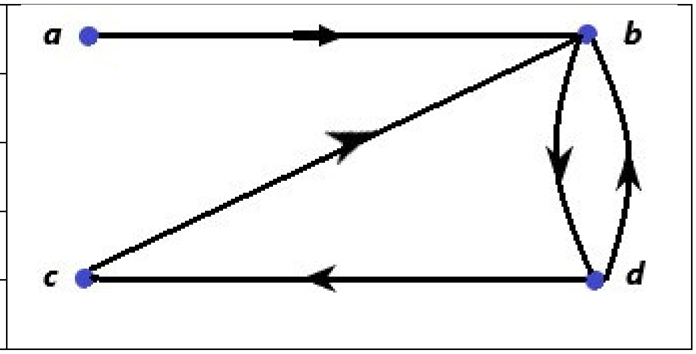
☺

3. Usmereni grafovi

Usmereni graf, ili *digraf*, je par *G* = (*A*, *L*) gde je *A* konačan skup čvorova, a *L* je relacija definisana na *A*. Vizuelna reprezentacija digrafa se sastoji od označenih čvorova koji su povezani usmerenim lukovima. Jednostavan digraf nema ni petlje ni višestruke lukove. To znači da postoji najviše jedan luk *ab* koji povezuje čvor *a* sa čvorom *b* i najviše jedan luk koji povezuje čvor *b* sa čvorom *a*.

Na slici je prikazan jedan usmereni jednostavni graf koji se sastoji od čvorova *A* = {*a*, *b*, *c*, *d*} i skupa lukova *L* = {*ab*, *bd*, *cb*, *db*, *dc*}, kao i matrica susedstva koja sada nije simetrična (jer ni relacija nije simetrična).

	a	*b*	*c*	*d*
a	F	T	F	F
b	F	F	F	T
c	F	T	F	F
d	F	T	T	F



Put dužine k u jednom digrafu je niz različitih čvorova $a_0, a_1, ..., a_k$ takvih da su $a_{i-1}\ a_i$ usmereni lukovi grafa za $i = 1, ..., k$. Jedan takav put se označava sa $a_0a_1...a_k$. Na primer, u gornjem zadatku je *a b d c* jedan put dužine 3.

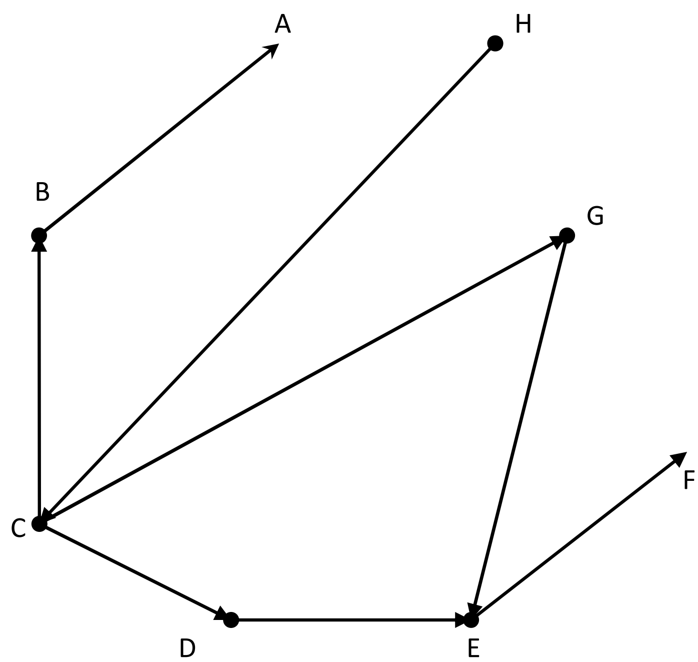
Ciklus u grafu je niz čvorova $a_0, a_1, ..., a_k$ takvih da su $a_{i-1}\ a_i$ usmereni lukovi grafa za $i = 1, ..., k$ i da je $a_0 = a_k$, pri čemu se nijedan drugi čvor ne ponavlja. Na primer, u gornjem zadatku je *b d c b* jedan ciklus. *Aciklični grafovi* su grafovi koji ne sadrže nijedan ciklus.

Aciklični grafovi su korisni za modeliranje situacija u kojima neki zadatak treba da se obavi pre nekog drugog. Postojanje ciklusa u grafu bi impliciralo da se neki zadatak obavlja pre samog sebe, što je nelogično. Kod problema uređivanja zadataka, odgovarajući aciklični graf se naziva PERT dijagram (eng. *Project Evaluation and Review Technique*).

Zadatak 6. U toku studija biologije student želi da uključi određene predmete, ali je za svaki od njih potreban neki preduslov. Student treba da odredi kojim redosledom da pohađa i polaže ove predmete – u tome mu pomaže PERT dijagram.

Primena biotehnologija (A)	B
Opšta biotehnologija (B)	C
Molekularna biologija (C)	H
Struktura DNK (D)	C
Enzimi (E)	D, G
Tehnologija hrane (F)	E
Genetsko inženjerstvo (G)	C
Humana bilogija (H)	ništa

PERT dijagram ovih zadataka bi izgledao ovako:



Ako student želi da utvrdi redosled kojim treba da pristupi predmetima da bi bili zadovoljeni preduslovi za svaki od njih, do toga može doći sledeći algoritam koji se naziva *topološko uređivanje*. Ovaj algoritam koherentno označava sve čvorove u bilo kom acikličnom grafu s n čvorova, 1, 2, 3..., n, tako da ako je uv jedan luk u grafu i i je redosled čvora u, a j redosled čvora v, onda je $i < j$, tj. čvor u prethodi čvoru v.

Algoritma topološkog uređivanja. Ovaj algoritam dozvoljava da se dobije koherentno označavanje acikličnog grafa $G = (V, E)$. Na početku su svi prethodnici čvora v sadržani u skupu $A(v)$.

početak

  za $v \in V$ **ponavljaj** *izračunaj* $A(v)$

  oznaka = 0

  sve dok postoje neoznačeni čvorovi v za koje je $A(v) = \varnothing$ **ponavljaj**

  početak

    oznaka = *oznaka* + 1

    u je jedan čvor za koji je $A(u) = \varnothing$

    označi čvor u sa *oznaka*

    za svaki neoznačeni čvor $v \in V$ **ponavljaj** $A(v) = A(v) - \{u\}$

  kraj

kraj

Rad algoritma se za aciklični graf iz zadatka 6. odvija u sledećim koracima:

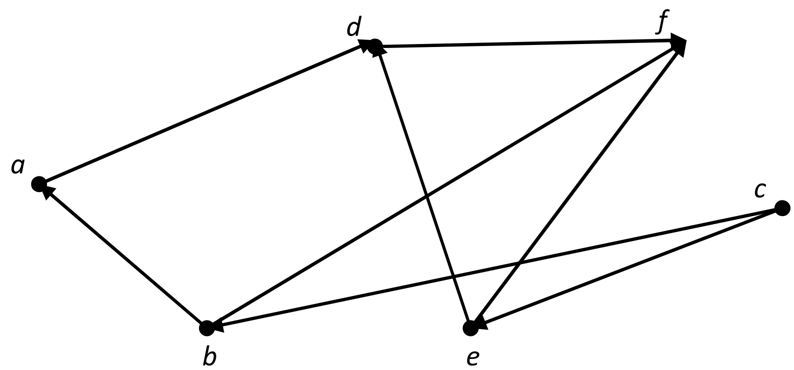
- Korak 0. (**za** $v \in V$ **ponavljaj** *izračunaj* $A(v)$): $A(\mathbf{A}) = \{\mathbf{B}\}$, $A(\mathbf{B}) = \{\mathbf{C}\}$, $A(\mathbf{C}) = \{\mathbf{H}\}$, $A(\mathbf{D}) = \{\mathbf{C}\}$, $A(\mathbf{E}) = \{\mathbf{D}, \mathbf{G}\}$, $A(\mathbf{F}) = \{\mathbf{E}\}$, $A(\mathbf{G}) = \{\mathbf{C}\}$, $A(\mathbf{H}) = \varnothing$
- Korak 1. Ulazak u petlju **sve dok**. Samo čvor **H** zadovoljava uslov $A(\mathbf{H}) = \varnothing$, pa čvor **H** dobija oznaku 1. Ovaj čvor se briše iz svih skupova $A(v)$: $A(\mathbf{A}) = \{\mathbf{B}\}$, $A(\mathbf{B}) = \{\mathbf{C}\}$, $A(\mathbf{C}) = \varnothing$, $A(\mathbf{D}) = \{\mathbf{C}\}$, $A(\mathbf{E}) = \{\mathbf{D}, \mathbf{G}\}$, $A(\mathbf{F}) = \{\mathbf{E}\}$, $A(\mathbf{G}) = \{\mathbf{C}\}$.
- Korak 2. Drugi prolaz kroz petlju **sve dok**. Samo čvor **C** zadovoljava uslov $A(\mathbf{C}) = \varnothing$, pa čvor **C** dobija oznaku 2. Ovaj čvor se briše iz svih skupova $A(v)$, gde je v neoznačeni čvor: $A(\mathbf{A}) = \{\mathbf{B}\}$, $A(\mathbf{B}) = \varnothing$, $A(\mathbf{D}) = \varnothing$, $A(\mathbf{E}) = \{\mathbf{D}, \mathbf{G}\}$, $A(\mathbf{F}) = \{\mathbf{E}\}$, $A(\mathbf{G}) = \varnothing$.
- Korak 3. Treći prolaz kroz petlju **sve dok**. Više čvorova zadovoljava uslov $A(v) = \varnothing$, pa se može izabrati bilo koji od njih, a svaki će voditi ka drugom koherenntnom redosledu. Izaberimo čvor **B** i dodelimo mu oznaku 3. Ovaj čvor se briše iz svih skupova $A(v)$, gde je v neoznačeni čvor: $A(\mathbf{A}) = \varnothing$, $A(\mathbf{D}) = \varnothing$, $A(\mathbf{E}) = \{\mathbf{D}, \mathbf{G}\}$, $A(\mathbf{F}) = \{\mathbf{E}\}$, $A(\mathbf{G}) = \varnothing$.
- Korak 4. Četvrti prolaz kroz petlju **sve dok**. Više čvorova zadovoljava uslov $A(v) = \varnothing$. Izaberimo čvor **A** i dodelimo mu oznaku 4. Ovaj čvor se briše iz svih skupova $A(v)$, gde je v neoznačeni čvor: $A(\mathbf{D}) = \varnothing$, $A(\mathbf{E}) = \{\mathbf{D}, \mathbf{G}\}$, $A(\mathbf{F}) = \{\mathbf{E}\}$, $A(\mathbf{G}) = \varnothing$.
- Korak 5. Peti prolaz kroz petlju **sve dok**. Izaberimo čvor **D** i dodelimo mu oznaku 5. Ovaj čvor se briše iz svih skupova $A(v)$, gde je v neoznačeni čvor: $A(\mathbf{E}) = \{\mathbf{G}\}$, $A(\mathbf{F}) = \{\mathbf{E}\}$, $A(\mathbf{G}) = \varnothing$.
- Korak 6. Šesti prolaz kroz petlju **sve dok**. Izaberimo čvor **G** i dodelimo mu oznaku 6. Ovaj čvor se briše iz svih skupova $A(v)$, gde je v neoznačeni čvor: $A(\mathbf{E}) = \varnothing$, $A(\mathbf{F}) = \{\mathbf{E}\}$.
- Korak 7. Sedmi prolaz kroz petlju **sve dok**. Izaberimo čvor **E** i dodelimo mu oznaku 7. Ovaj čvor se briše iz svih skupova $A(v)$, gde je v neoznačeni čvor: $A(\mathbf{F}) = \varnothing$.
- Korak 8. Poslednji prolaz kroz petlju **sve dok**. Čvoru **F** dodeljena oznaka 8.
- Rezultat: jedan koherentan redosled čvorova (kurseva koje treba slušati) je: **H, C, B, A, D, G, E, F**.

☺

Zadatak 7. Data je matrica jednog acikličnog grafa:

	a	b	c	d	e	f
a	F	F	F	T	F	F
b	T	F	F	F	F	T
c	F	T	F	F	T	F
d	F	F	F	F	F	T
e	F	F	F	T	F	T
f	F	F	F	F	F	F

(a) Nacrtati usmereni graf;
(b) Primeniti algoritam topološkog uređivanja;
(c) Prepisati tablicu tako da kolone i vrste budu uređene u skladu s dobijenim koherentnim uređivanjem.



(b) Korak 0. Početni skupovi prethodnika svih čvorova su $A(\mathbf{a}) = \{\mathbf{b}\}$, $A(\mathbf{b}) = \{\mathbf{c}\}$, $A(\mathbf{c}) = \varnothing$, $A(\mathbf{d}) = \{\mathbf{a, e}\}$, $A(\mathbf{e}) = \{\mathbf{c}\}$, $A(\mathbf{f}) = \{\mathbf{b, d, e}\}$.
Korak 1: Biramo čvor c i dodeljujemo mu indeks 1, pa ga brišemo iz svih skupova prethodnika: $A(\mathbf{a}) = \{\mathbf{b}\}$, $A(\mathbf{b}) = \varnothing$, $A(\mathbf{d}) = \{\mathbf{a, e}\}$, $A(\mathbf{e}) = \varnothing$, $A(\mathbf{f}) = \{\mathbf{b, d, e}\}$.
Korak 2: Biramo, recimo, čvor b i dodeljujemo mu indeks 2, pa ga brišemo iz svih skupova prethodnika: $A(\mathbf{a}) = \varnothing$, $A(\mathbf{d}) = \{\mathbf{a, e}\}$, $A(\mathbf{e}) = \varnothing$, $A(\mathbf{f}) = \{\mathbf{d, e}\}$.
Korak 3: Biramo, recimo, čvor a i dodeljujemo mu indeks 3, pa ga brišemo iz svih skupova prethodnika: $A(\mathbf{d}) = \{\mathbf{e}\}$, $A(\mathbf{e}) = \varnothing$, $A(\mathbf{f}) = \{\mathbf{d, e}\}$.
Korak 4: Biramo čvor e i dodeljujemo mu indeks 4, pa ga brišemo iz svih skupova prethodnika: $A(\mathbf{d}) = \varnothing$, $A(\mathbf{f}) = \{\mathbf{d}\}$.
Korak 5: Biramo čvor d i dodeljujemo mu indeks 5, pa ga brišemo iz svih skupova prethodnika: $A(\mathbf{f}) = \varnothing$.
Korak 5: Na kraju je $A(\mathbf{f}) = \varnothing$ i čvor f dobija indeks 6.
Koherentno uređivanje čvorova je: $c(1)$, $b(2)$, $a(3)$, $e(4)$, $d(5)$, $f(6)$.
Nova tablica susedstva je:

	c	b	a	e	d	f
c	F	T	F	T	F	F
b	F	F	T	F	F	T
a	F	F	F	F	T	F
e	F	F	F	F	T	T
d	F	F	F	F	F	T
f	F	F	F	F	F	F

☺

Zadatak 8. Data je tabela zadataka u kojoj je za svaki zadatak naznačeno koji zadaci treba prethodno da budu uređeni. Uredi ove zadatke koristeći algoritam topološkog uređivanja tako da se mogu redom obaviti.

A	iseckaj luk	I
B	operi salatu	K
C	pripremi sos	K
D	počni prženje	J
E	pomešaj salatu	B, C
F	iseci piletinu	ništa
G	izrendaj đumbir	I
H	fino naseckaj kupus	I
I	mariniraj piletinu	F
J	zagrej vok	A, G, H, K
K	pripremi pirinač	ništa

☺

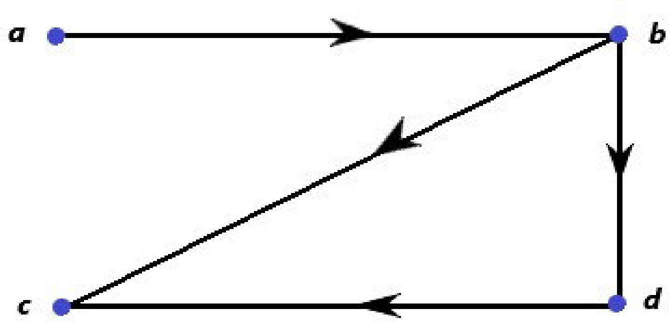
Usmereni grafovi se koriste za modeliranje raznih stvari, na primer, za modeliranje avionskih letova. Na primer, ako neki avion ne može da sletii na neki aerodrom, treba znati da li postoji neki drugi put kojim može ići. Slično, ako je više puteva presečeno u nekoj informatičkoj mreži, neki korisnici mogu da budu odsečeni. S toga je potrebno znati da li u nekom grafu postoji put koji povezuje bilo koja dva čvora u usmerenom grafu.

Ako je $G = (A, L)$ digraf koji ima n čvorova i M je njegova matrica susedstva, onda vrednost T u matrici ukazuje da postoji usmereni put dužine 1 u grafu G. Na šta ukazuje proizvod $M \cdot M$? Vrednost T u ovoj matrici će ukazivati na postojanje puta dužine dva između dva čvora. Kako se izračunava matrica $M \cdot M$?

Vrednost $m_{\_2\_ij}$ u ovoj matrici se dobija kao disjunkcija konjunkcija odgovarajućih vrednosti vrste *i* matrice *M* i kolone *j* iste matrice. Na primer, vrednost $m_{\_2\_aa}$ za graf iz sledećeg zadatka se izračunava kao $m_{\_2\_aa}$ = (F ∧ F) ∨ (T ∧ F) ∨ (F ∧ F) ∨ (F ∧ F) = F ∨ F ∨ F ∨ F = F, što znači da ne postoji put dužine 2 iz čvora *a* u čvor *a*. Vrednost $m_{\_2\_ad}$ se izračunava kao $m_{\_2\_ad}$ = (F ∧ F) ∨ (T ∧ T) ∨ (F ∧ F) ∨ (F ∧ F) = F ∨ T ∨ F ∨ F = T, što znači da postoji put dužine 2 iz čvora *a* u čvor *d* (preko čvora *b*).

Na sličan način je $M^3 = M^2 \cdot M$, i ova matrica predstavlja sve puteve dužine 3 u polaznom grafu. Konačno $M^* = M \vee M^2 \vee M^3 \vee \ldots \vee M^n$ je matrica koja predstavlja sve puteve u grafu proizvoljne dužine. Operacija disjunkcije na matricama se obavlja tako što se obavi disjunkcija odgovarajućih elemenata u matricama.

Zadatak 9. Odrediti koji sve putevi postoje u digrafu sa slike računanjem matrice *M**.



$$M = \begin{vmatrix} F & T & F & F \\ F & F & T & T \\ F & F & F & F \\ F & F & T & F \end{vmatrix} \quad M \cdot M = \begin{vmatrix} F & T & F & F \\ F & F & T & T \\ F & F & T & T \\ F & F & T & T \end{vmatrix} \cdot \begin{vmatrix} F & T & F & F \\ F & F & T & T \\ F & F & T & T \\ F & F & T & T \end{vmatrix} = \begin{vmatrix} F & F & T & T \\ F & F & T & F \\ F & F & F & F \\ F & F & T & F \end{vmatrix}$$

$$M^3 = M^2 \cdot M = \begin{vmatrix} F & F & T & T \\ F & F & T & F \\ F & F & F & F \\ F & F & T & F \end{vmatrix} \cdot \begin{vmatrix} F & T & F & F \\ F & F & T & T \\ F & F & T & T \\ F & F & T & T \end{vmatrix} = \begin{vmatrix} F & F & T & F \\ F & F & F & F \\ F & F & F & F \\ F & F & F & F \end{vmatrix}$$

$$M^4 = M^3 \cdot M = \begin{vmatrix} F & F & T & F \\ F & F & F & F \\ F & F & F & F \\ F & F & F & F \end{vmatrix} \cdot \begin{vmatrix} F & T & F & F \\ F & F & T & T \\ F & F & T & T \\ F & F & T & T \end{vmatrix} = \begin{vmatrix} F & F & F & F \\ F & F & F & F \\ F & F & F & F \\ F & F & F & F \end{vmatrix}$$

$$M^* = \begin{vmatrix} F & T & T & T \\ F & F & T & T \\ F & F & F & F \\ F & F & T & F \end{vmatrix}$$

Iz ovoga se vidi da u grafu postoji samo jedan put dužine 3 – *ab bd dc* - a da puteva dužine 4 nema. Takođe se iz M^* vidi da se iz čvora b ne može doći u čvor a, iz čvora c se ne može stići ni u jedan čvor, dok se iz čvora d može stići samo u čvor c.

☺

Kako bi se mogla izračunati ova matrica?

Algoritam grube sile za računanje svih puteva u grafu

početak

M_0 je ulazna matrica grafa koji ima m čvorova

$n = 1$

sve dok je u $M_{n\text{-}1}$ bar jedna vrednost $\neq$ F (tj. V = T) **ponavljaj**

  $n = n + 1$ /* računaju se razni stepeni */

  za sve i = 1 **do** m /* prolazi se kroz vrste matrice */

    V = F

    za sve j = 1 **do** m /* prolazi se kroz kolone matrice */

      $M_n(i, j)$ = F

      za sve k = 1 **do** m /* računa se se jedna vrednost u matrici M_n */

        $M_n(i, j) = M_n(i, j) \vee (M_{n\text{-}1}(i, k) \wedge M(k, j))$

      $V = V \vee M_n(i, j)$

    kraj za sve /* j */

  kraj za sve /* i */

kraj sve_dok

/* računaj M^* */

za sve i = 1 **do** m /* prolazi se kroz vrste matrice */

  za sve j = 1 **do** m /* prolazi se kroz kolone matrice */

    $M^*(i, j)$ = F

      za sve k = 1 **do** $n\text{-}1$/* računa se za sve stepene M_n */

        $M^*(i, j) = M^*(i, j) \vee M_k(i, j)$

kraj

Ovo je suviše neefikasan postupak. Srećom postoji algoritam koji Matricu M^* računa na mnogo efikasniji način;

Voršalov algoritam: Ovaj algoritam računa matricu dostupnosti $W = M^*$ nekog digrafa $G = (V, E)$ čija je matrica susedstva M.

početak

$W = M$

za sve $k = 1$ **do** m /* stepeni matrice */

za sve $i = 1$ **do** m /* prolazi se kroz redove matrice */

za sve $j = 1$ **do** m /* prolazi se kroz kolone */

$W(i, j) = W(i, j) \vee (W(i, k) \wedge W(k, j))$

kraj

Objašnjenje algoritma;

U svakom prolazu kroz spoljnju petlju indeksiranu sa k algoritam računa matricu W_k polazeći od koeficijenata matrice W_{k-1}.

Da bi se izačunao red i matrice W_k, treba da se izračuna izraz;

$W(i, j) \vee (W(i, k) \wedge W(k, j))$

za razne vrednosti j.

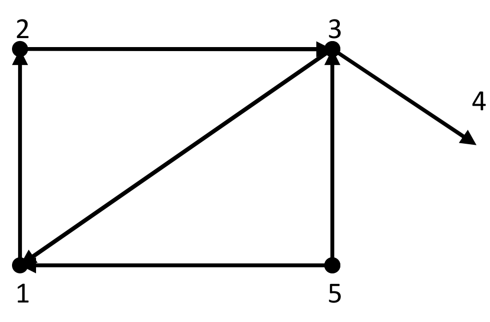
Ako je $W(i, k)$ = F onda je $W(i, k) \wedge W(k, j)$=F pa se ceo izraz svodi na vrednost $W(i, j)$. To znači da red i matrice ostaje nepromenjen.

Ako je $W(i, k)$ = T, onda se ceo izraz svodi na $W(i, j) \vee (T \wedge W(k, j)) = W(i, j) \vee W(k, j)$, U tom slučaju se računanje reda i svodi na disjunkciju redova i i k.

Kako se, dakle računa W_k polazeći od W_{k-1}.

1. Razmatra se kolona k matrice W_{k-1};
2. Svaki red koji u toj koloni ima vrednost F se prepisuje u matricu W_k;
3. Za svaki red koji u toj koloni ima vrednost T, red u matrici W_k. se dobija disjunkcijom tog reda i reda k.

Primer; Neka je dat graf sa slike:



Matrica susedstva ovog grafa je:

$$W_0 = \begin{vmatrix} F & T & F & F & F \\ F & F & T & F & F \\ T & F & F & T & F \\ F & F & F & F & F \\ T & F & T & F & F \end{vmatrix}$$

Da bismo izračunali W_k. u koraku 1 treba da uzmemo u razmatranje kolonu 1 matrice W_0. U koraku 2 treba da prepišemo redove 1, 2, 4 direktno u matricu W_1 jer su odgovarajuće vrednosti u koloni 1 F:

$$W_1 = \begin{vmatrix} F & T & F & F & F \\ F & F & T & F & F \\ & & & & \\ F & F & F & F & F \\ & & & & \end{vmatrix}$$

U koraku 3 računamo disjunkciju redova 3 i 1 matrice W_0 da bismo dobili red 3 matrice W_1:

$$W_1 = \begin{vmatrix} F & T & F & F & F \\ F & F & T & F & F \\ T & T & F & T & F \\ F & F & F & F & F \\ & & & & \end{vmatrix}$$

To isto ponavljamo i za red 5 (disjunkcija redova 5 i 1 matrice W_0

$$W_1 = \begin{vmatrix} F & T & F & F & F \\ F & F & T & F & F \\ T & T & F & T & F \\ F & F & F & F & F \\ T & T & T & F & F \end{vmatrix}$$

Sada se nastavlja sa računanjem matrice W_2. Postupa se na isti način. Posmatra se kolona 2 matrice W_1. Prepisuju se redovi 2 i 4, a redovi 1, 3 i 5 se dobijaju disjunkcijom redova 1 i 2, 3 i 2, 5 i 2:

$$W_2 = \begin{vmatrix} F & T & T & F & F \\ F & F & T & F & F \\ T & T & T & T & F \\ F & F & F & F & F \\ T & T & T & F & F \end{vmatrix}$$

Na sličan način se računa i W_3. Postupa se na isti način. Posmatra se kolona 3 matrice W_2. Prepisuje se red 4, a redovi 1, 2, 3 i 5 se dobijaju disjunkcijom redova tih redova i reda 3:

$$W_3 = \begin{vmatrix} T & T & T & T & F \\ T & T & T & T & F \\ T & T & T & T & F \\ F & F & F & F & F \\ T & T & T & T & F \end{vmatrix}$$

Kako iz čvora 4 ne izlazi ni jedan luk, natrica W_4 je identična kao i matrica W_3 , a iz istog razloga je i matrica W_5 ista, pa je ovo i konačna matrica M^*. Naime, u koloni matrice W_3 je F samo u redu 4, pa se taj red prepisuje, a ostali redovi se dobijaju disjunkciojom sa redom 4. Kako su u ovom redu sve vrednosti F, to znači da će svi ostali redovi ostati nepromenjeni, tj. $W_3 = W_4 = W_5$.

☺

Zadatak 10. Digraf G ima matricu susedstva sledeću:

$$M = \begin{vmatrix} F & T & F & F \\ F & F & T & F \\ F & F & F & T \\ T & F & F & F \end{vmatrix}$$

(a) Izračunati matrice M^2, M^3, i M^4, i na osnovu toga odrediti šta bi bila matrica M^*;

(b) Koristeći Voršalov dijagram izračunati matrice W_1, W_2, W_3, i W_4,, i na osnovu toga odrediti šta bi bila matrica M^*;