Teorija baza podataka
Polustrukturirane baze podataka i podatkovni grafovi

Izv. prof. dr. sc. Markus Schatten

Fakultet organizacije i informatike,
Sveučilište u Zagrebu
Pavlinska 2, 42000 Varaždin
markus.schatten@foi.hr

Uvod

- Razvojem World Wide Weba, a posebice elektroničkog poslovanja došlo je do potrebe razmjene sve veće količine podataka putem mreže.
- Obzirom da je riječ o heterogenim izvorima podataka (kako u pogledu konceptualne tako i u pogledu logičke izvedbe) koji su često kompleksni i nepotpuni, razvijen je polustrukturirani model podataka.

Prednosti i nedostaci polustrukturiranog modela podataka

- Prednosti polustrukturiranog modela su
  - što je u mogućnosti reprezentirati podatke koje nije moguće ograničiti shemom (ponekad se prikazuje kao model podataka bez sheme, odnosno samo-opisujući model podataka),
  - fleksibilan je u pogledu razmjene podataka,
  - ponekad je korisno razmatrati strukturirane podatke kao polustrukturirane (posebice pri pregledavanju na Webu),
  - shemu podataka lako je mijenjati te
  - je format zapisa prenosiv.
- Najveći nedostatak polustrukturiranog modela, koji se javlja upravo zbog fleksibilnosti u pogledu strukture, jest nemogućnost postizanja efikasnosti postavljanja upita kao što je to slučaj kod relacijskih i drugih bolje strukturiranih modela podataka.

Polustrukturirani model podataka

Za predstavljanje polustrukturiranog modela podataka koristit ćemo teoriju grafova.

Definicija

Usmjereni graf

Neka je V skup vrhova (čvorova) i neka je $B \subseteq V \times V$ skup bridova (skup uređenih parova vrhova). Usmjereni graf G je uređeni par (V, B). Za svaki brid $b = (v_i, v_j)$ kažemo da je v_i izvorište brida ($izv(b)$), a v_j odredište brida ($odr(b)$).

Put

Definicija

Put

Neka je $G = (V, B)$ usmjereni graf. Svaki niz bridova $b_1/b_2/\ldots/b_k$ za koje vrijedi $odr(b_i) = izv(b_{i+1}), i = 1, 2, \ldots, k-1$ kažemo da je put u grafu G odnosno put od izvorišta $izv(b_1)$ do odredišta $odr(b_k)$. Broj bridova u putanji, k, naziva se duljinom puta.

Korijen

Definicija

Korijen

Za vrh v_k kažemo da je korijen usmjerenog grafa $G = (V, B)$, ako postoji put od v_k do svakog vrha $v_i \in V, i \neq k$.

Ciklus

Definicija

Ciklus

Ciklus u usmjerenom grafu je svaki put od nekog vrha k samom sebi. Graf bez ciklusa naziva se acikličnim.

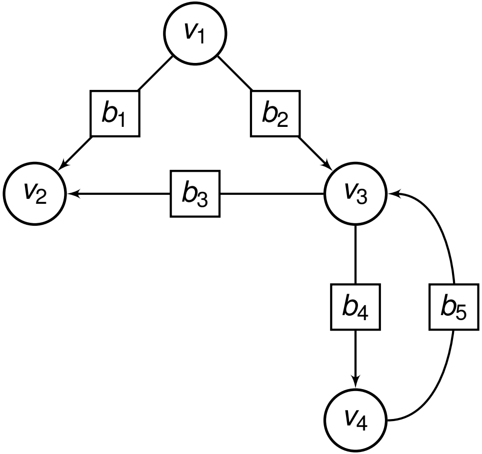
Primjer I

Neka je zadan usmjereni graf $G = (V, B)$, $V = \{v_1, v_2, v_3, v_4\}$, $B = \{b_1, b_2, b_3, b_4, b_5\}$ i neka je:

brid	izvorište	odredište
b_1	v_1	v_2
b_2	v_1	v_3
b_3	v_3	v_2
b_4	v_3	v_4
b_5	v_4	v_3

Graf G prikazuje se grafički na sljedeći način:

Primjer II



Primjer III

Nekoliko puteva u G: b_1 ; b_2/b_3 ; $b_2/b_4/b_5$; $b_5/b_4/b_5/b_4/b_5$; $b_2/b_4/b_5/b_3$
Korijen od G: v_1
Nekoliko ciklusa u G: b_4/b_5 ; b_5/b_4 ; $b_4/b_5/b_4/b_5$
U nastavku ćemo radi preglednosti grafove zadavati grafički.

Stablo

Definicija

Stablo

Usmjereni graf (V, B) je stablo akko postoji jedinstveni put od v_k do v_i za svaki $v_i \in V, i \neq k$. Uočite da je svako stablo nužno aciklično i da ima jedinstveni korjen.

List

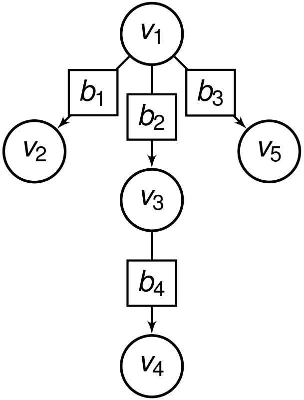
Definicija

List

Vrh $v \in V$ je list usmjerenog grafa (V, B) ako ne postoji niti jedan $b \in B$ takav da $izv(b) = v$.

Primjer

Neka je zadan aciklički usmjereni graf G:



Uočite da je G stablo. Vrhovi v_2, v_4 i v_5 su listovi. Vrh v_1 je korijen stabla G.

Podatkovni graf

Usmjerene grafove možemo iskoristiti za prikaz podataka, te ćemo ih u tom kontekstu nazivati podatkovim grafovima. Podatkovni grafovi najčešće su aciklički ali postoje iznimke.

Definicija

Podatkovni graf

Podatkovni graf je usmjereni graf $G_P = (V, B)$ čiji su bridovi označeni (oznaka $o \sim b$, $b \in B$), a vrhovi podatkovni objekti koji mogu biti:

1. *atomarni (listovi)*
2. *složeni (imaju bridove k drugim vrhovima)*

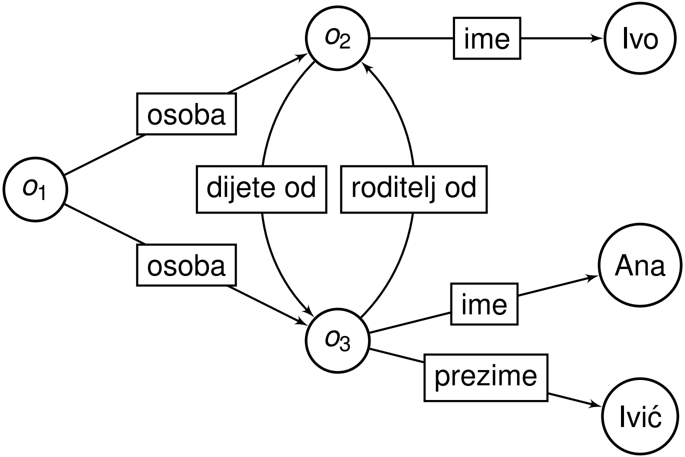
Svaki složeni vrh ima svoj identitet objekta koje je jedinstven u čitavom grafu. Ponekad je korisno razmatrati skup vrhova kao uniju $V = V_a \cup V_s$, pri čemu je V_a skup identiteta atomarnih objekata, a skup V_s skup identiteta složenih objekata i vrijedi $V_a \cap V_s = \varnothing$.

Napomena

U nastavku ćemo razmatrati samo podatkovne grafove koji imaju određeni korijen. U skladu s time možemo podatkovne grafove definirati kao uređenu trojku $G_P = (V_a \cup V_s, B, k)$, pri čemu je $k \in V_a \cup V_s$ korijen od G_P.

Primjer I

Razmotrimo sljedeći podatkovni graf G_1:

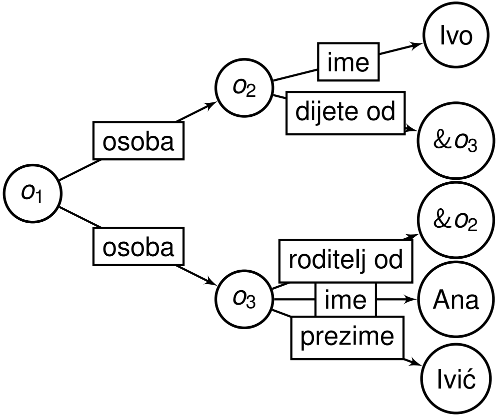


Primjer II

U grafu smo identitete podatkovnih objekata označili sa o_1, o_2, o_3. Vrhovi Ivo, Ana i Ivić su atomarni, tj. $V_a = \{\text{Ivo}, \text{Ana}, \text{Ivić}\}$. Vrhovi o_1, o_2 i o_3 su složeni, tj. $V_s = \{o_1, o_2, o_3\}$. Bridovi ime i dijete od čvora o_2 označavaju da on ima veze odgovarajuće oznake k drugim podatkovnim objektima. Uočite da graf G_1 nije stablo.

Graf smo mogli predstaviti i na sljedeći način:

Primjer III



U ovom smo grafu identitete objekata iskoristili kao reference (prefiks &), čime smo graf G_1 pretvorili u pseudo-stablo.

Primjer I

Podatkovne grafove moguće je iskoristiti za prikaz različitih vrsta podataka.

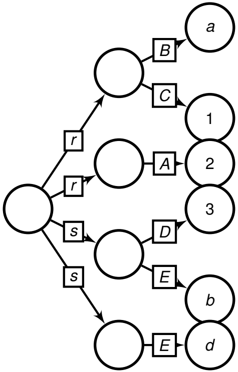
Neka je zadana sljedeća (parcijalna) relacijska baza podataka:

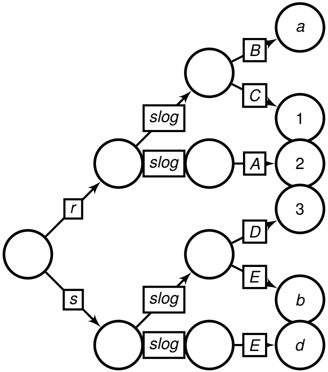
r	A	B	C
	?	a	1
	2	?	?

s	D	E
	3	b
	?	d

Za prikaz ove baze podataka moguće je odabrati različite oblike grafova (radi jednostavnosti su ispušteni identiteti objekata):

Primjer II





Učite da se podatkovni grafovi mogu iskoristiti za prikaz nepotpunih podataka.

Simulacija

Uvodimo pojam simulacije koja je osnova za graf sheme.

Definicija

Simulacija

Neka su $G_1 = (V_1, B_1, k_1)$ i $G_2 = (V_2, B_2, k_2)$ dva podatkovna grafa s određenim korijenima. Relacija ς nad V_1, V_2 je simulacija ako zadovoljava sljedeće uvjete:

1. $k_1 \varsigma k_2$; *i*
2. $\forall x_1, y_1 \in V_1, \forall x_2 \in V_2 (l = (x_1, y_1) \in B_1 \land x_1 \varsigma x_2 \Rightarrow \exists y_2 \in V_2 (l = (x_2, y_2) \in B_2 \land (y_1 \varsigma y_2)))$

Simulacija

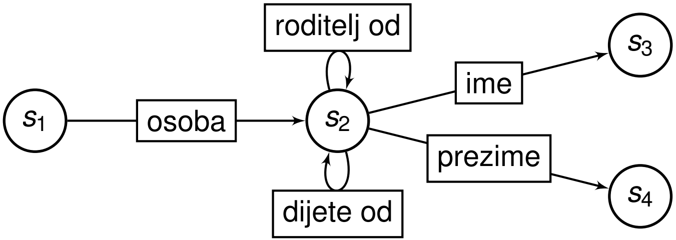
Neformalno, simulacija podatkovnog grafa G_1 na graf G_2 znači da ako postoji brid u G_1, postoji odgovarajući brid jednake oznake u G_2. Podatkovni graf G_P je instanca grafa sheme S, notacija $G_P \trianglelefteq S$, ako postoji simulacija od G_P na S.

Propozicija

Relacija simulacije je tranzitivna.

Primjer I

Neka je zadan graf S_1.



Primjer II

Uočite da je graf G_1 iz prethodnog primjera instanca grafa S_1. To dokazuje simulacija s_1:

s_1	v_1	v_2	
	o_1	s_1	$\rightarrow$ korijen
	o_2	s_2	$\rightarrow$ jer je osoba $\sim (o_1, o_2)$; osoba $\sim (s_1, s_2) \wedge$ roditelj od $\sim (o_3, o_2)$; roditelj od $\sim (s_2, s_2)$
	o_3	s_2	$\rightarrow$ jer je osoba $\sim (o_1, o_3)$; osoba $\sim (s_1, s_2) \wedge$ dijete od $\sim (o_2, o_3)$; dijete od $\sim (s_2, s_2)$
	Ivo	s_3	$\rightarrow$ jer je ime $\sim (o_2, \text{Ivo})$; ime $\sim (s_2, s_3)$
	Ana	s_3	$\rightarrow$ jer je ime $\sim (o_3, \text{Ana})$; ime $\sim (s_2, s_3)$
	Ivić	s_4	$\rightarrow$ jer je prezime $\sim (o_2, \text{Ivić})$; prezime $\sim (s_2, s_4)$

Kontrakcija čvorova

Za konstrukciju grafa sheme iz prethodnog primjera poslužili smo se višestrukom upotrebom operacije kontrakcije čvorova s jednako označenim ulaznim bridovima.

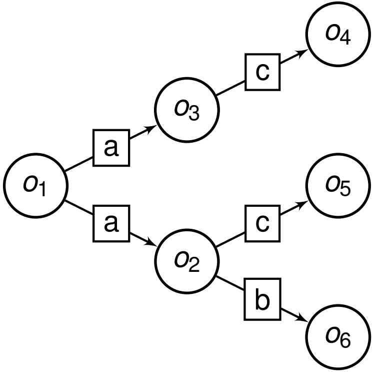
Definicija

Kontrakcija čvorova

Neka je zadan podatkovni graf $G = (V, B, k)$ i neka su $v_1, v_2 \in V$ dva čvora iz G čiju su ulazni bridovi b_1, b_2 jednako označeni s oznakom o. Kontrakcija čvorova v_1, v_2 u grafu G oznaka $\pitchfork_b [v_1, v_2](G)$ je usmjereni graf u kojem su vrhovi v_1, v_2 zamjenjeni jedinstvenim vrhom v, a bridovi b_1, b_2 jedinstvenim bridom b s oznakom o. Za vrh v vrijedi da je izvorište svih bridova za koje je vrijedilo $izv(v_1) \vee izv(v_2)$, te je odredište svih bridova za koje je vrijedilo $odr(v_1) \vee odr(v_2)$.

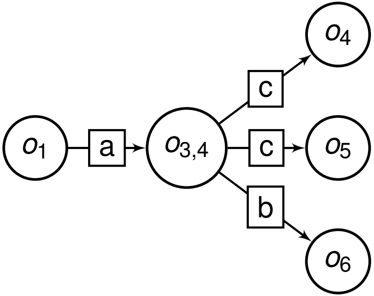
Primjer I

Neka je zadan podatkovni graf G:



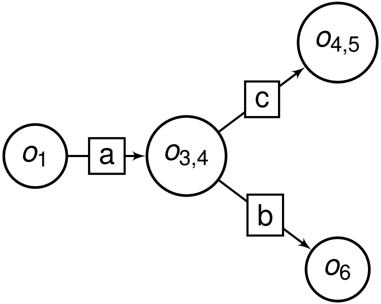
Kontrakcija $G_1 = \pitchfork_a [o_2, o_3](G)$ zadana je grafom:

Primjer II



Kontrakcija $G_2 = \pitchfork_c [o_4, o_5](G_1)$ zadana je grafom:

Primjer III



Propozicija

Propozicija

Podatkovni graf dobiven kontrakcijom čvorova simulacija je originalnog grafa.

Dokaz.

Neka je G podatkovni graf i neka je $G_k = \pitchfork_b [c_1, c_2](G)$ graf dobiven kontrakcijom čvorova c_1, c_2 preko oznake b. Neka je c_k identitet novonastalog čvora. Neka je relacija s definirana na sljedeći način: svaki čvor iz G je u relaciji sam sa sobom osim čvorova c_1 i c_2 koji su u relaciji sa čvorom c_k. Vidimo da je relacija s simulacija G na G_k. □

Pronalazak minimalnog grafa sheme

Algoritam

(Pronalaženje minimalnog grafa sheme)

Ulaz: podatkovni graf $G = (V, B, k)$
Izlaz: minimalni graf S koji ima svojstvo da je G instanca od S

1. *Staviti* $S \leftarrow G$.
2. $\forall b \in B$ *ako* $\exists b' \in B$ *koji ima istu oznaku o staviti* $S \leftarrow \pitchfork_o [odr(b), odr(b')](S)$.
3. *Ako više nema dva brida s jednakim oznakama tada je S minimalni graf sheme.*

Ograničenje domene lista

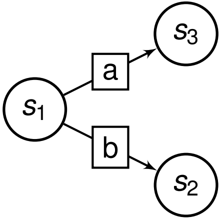
Najjednostavnije ograničenje nad polustrukturiranim podacima je ograničenje domene listova u grafu sheme.

Definicija

Ograničenje domene lista
Neka je zadan graf sheme $S = (V, B, k)$ i neka je $s_l \in V$ jedan od listova grafa. Ograničenje $Dom(s_l) = D$ je ograničenje domene nad listom s_l. Da bi neki podatakovni graf zadovoljavao graf sheme mora vrijediti da odgovarajući listovi poprimaju vriednosti isključivo iz skupa D.

Primjer I

Neka je zadan graf sheme S:

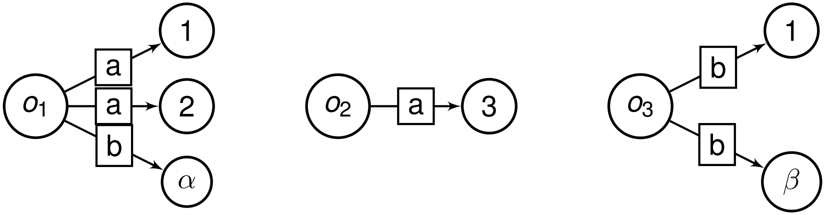


Neka su nad grafom sheme zadana sljedeća ograničenja domene:

$$
\begin{aligned}
Dom(s_2) &= \{1, 2, 3\} \\
Dom(s_3) &= \{\alpha, \beta\}
\end{aligned}
$$

Primjer II

Promotrimo sljedeće podatkovne grafove (G_1, G_2 i G_3 respektivno):



Grafovi G_1 i G_2 su instance od S. Graf G_3 nije instanca od S jer ne zadovoljava ograničenja domene.

Ogreničenje kardinalnosti

Definicija

Ograničenje kardinalnosti
Neka je zadan graf sheme $S = (V, B, k)$ i neka je o oznaka nekog brida u $b \in B$. Izrazi $Kard_{min}(o) = x$ i $Kard_{max}(o) = y$, $x \in \mathbb{N}_0; y \in \mathbb{N} \cup \{*\}$ označavaju ograničenja minimalne odnosno maksimalne kardinalnosti oznake o. Da bi neki podatkovni graf bio instanca grafa sheme S mora vrijediti da svi vrhovi koji imaju izlazne bridove s oznakom o imaju minimalno x i maksimalno y takvih bridova.

Oznaka $*$ označava “više” i znači da je maksimalna kardinalnost može poprimiti bilo koji prirodni broj.

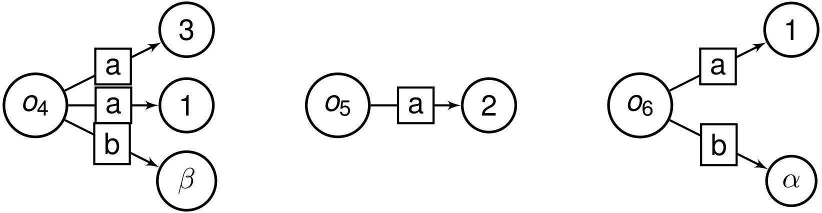
Primjer I

Neka je zadan graf sheme S kao u prethodnom primjeru i sljedeća ograničenja kardinalnosti:

$$\begin{aligned} Kard_{min}(a) &= 0 \\ Kard_{max}(a) &= * \\ Kard_{min}(b) &= 1 \end{aligned}$$

Primjer II

Promotrimo sljedeće podatkovne grafove (G_4, G_5 i G_6 respektivno):



Podatkovni grafovi G_4 i G_6 su instance grafa sheme S. Podatkovni graf G_5 nije instanca grafa sheme S jer ne zadovoljava ograničenje kardinalnosti $Kard_{min}(b) = 1$.

Upiti nad podatkovnim grafovima

U nastavku ćemo se za upite nad podatkovnim grafovima poslužiti sintaksom pravilnih izraza putanja.

Definicija

Abeceda

Abeceda Σ je konačan skup znakova.

Riječ i prazna riječ

Definicija

Riječ

Riječ iz abecede Σ je konačno polje od 0 ili više znakova iz Σ.

Definicija

Prazna riječ

Riječ s 0 znakova označava se sa ε i naziva se praznom riječju.

Skup riječi nad abecedom

Definicija

Skup riječi nad abecedom

Neka je Σ abeceda. Tada sa Σ^n, gdje je $n \geqslant 0$ označavamo skup svih riječi iz abecede Σ koje su dužine n. Stoga je skup svih riječi definiranih nad Σ:

$$\Sigma^* = \bigcup_{n \geqslant 0} \Sigma^n$$

Slično tome, skup svih nepraznih riječi nad Σ definira se kao:

$$\Sigma^+ = \bigcup_{n \geqslant 1} \Sigma^n$$

Sintaksa pravilnih izraza putanja

Definicija

Neka su zadani skup simbola $\mathbb{S} = \{\varepsilon, \_, +, *, ?, |, -\}$, *abeceda* $\Sigma = \Theta \cup \mathbb{S}$, Σ^* *skup svih riječi definiranih nad* Σ. *Definiramo pravilne izraze putanja kao najmanji skup* $\mathbb{PIP} \subseteq \Sigma^*$ *za koji vrijedi:*

1. $\forall s \in \Theta : s \in \mathbb{PIP}$;
2. $\varepsilon \in \mathbb{PIP}$ *(prazna riječ);*
3. $\_ \in \mathbb{PIP}$ *(bilo koji znak);*
4. $\forall i_1, i_2 \in \mathbb{PIP} : i_1 | i_2 \in \mathbb{PIP}$ *(alternacija);*
5. $\forall i_1, i_2 \in \mathbb{PIP} : i_1 i_2 \in \mathbb{PIP}$ *(konkatenacija);*
6. $\forall i \in \mathbb{PIP} : i? \in \mathbb{PIP}$ *(opcionalnost);*
7. $\forall i \in \mathbb{PIP} : i+ \in \mathbb{PIP}$ *(Kleene plus, jedno ili više ponavljanja);*
8. $\forall i \in \mathbb{PIP} : i* \in \mathbb{PIP}$ *(Kleene zvjezdica, nula ili više ponavljanja);*
9. $\forall s \in \Theta : s- \in \mathbb{PIP}$ *(negacija).*

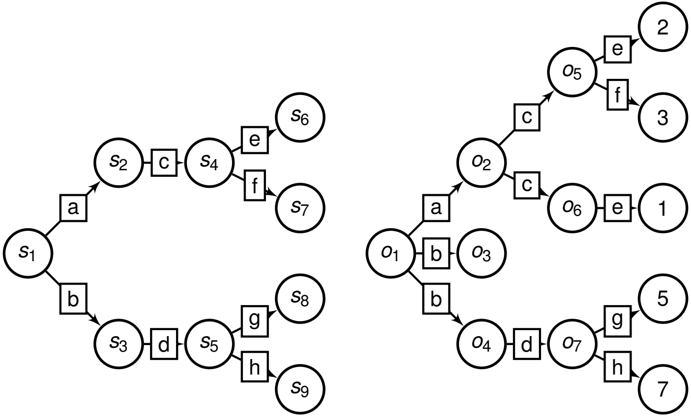
Napomene

Posebno za podatkovne grafove, abeceda je zadana kao $\Theta = \{a, b, c, a_1, a_2, \ldots\}$ skup oznaka definiranih nad odgovarajućim grafom sheme. Radi preglednosti uobičajeno je oznake odvajati znakom $/$ koji se može interpretirati kao čvor. Svaka se putanja interpretira od korijena (prvi znak $/$ predstavlja korijenski čvor).

Semantiku pravilnih izraza putanja prikazati ćemo na primjerima.

Primjer I

Neka je zadan graf sheme $S = (V_s, B_s, s_1)$ i njegova instanca $G = (V, B, o_1)$:



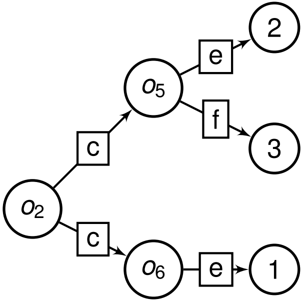
Primjer II

Iz grafa sheme vidimo da je abeceda $\Theta = \{a, b, c, d, e, f, g, h\}$. Razmotrimo sljedeće upite nad grafom G:

$$
\begin{aligned}
U_1 &: \quad /a \\
U_2 &: \quad /b \\
U_3 &: \quad \varepsilon \\
U_4 &: \quad /w
\end{aligned}
$$

Primjer III

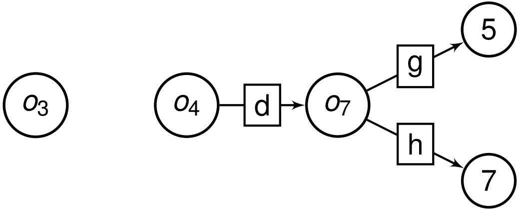
Rezultat upita U_1 je graf:



Vidimo da je upit označio sve čvorove koji zadovoljavaju putanju, a to je u ovom slučaju bio samo čvor o_2.

Primjer IV

Rezultat upita U_2 su grafovi:



Dakle, u ovom slučaju čvorovi o_3 i o_4 su zadovoljavali putanju $/b$.

Primjer V

Rezultat upita U_3 je prazan graf (oznaka $G^{\emptyset}$).

Upit U_4 nije primjenjiv na graf G jer $w \notin \Theta$.

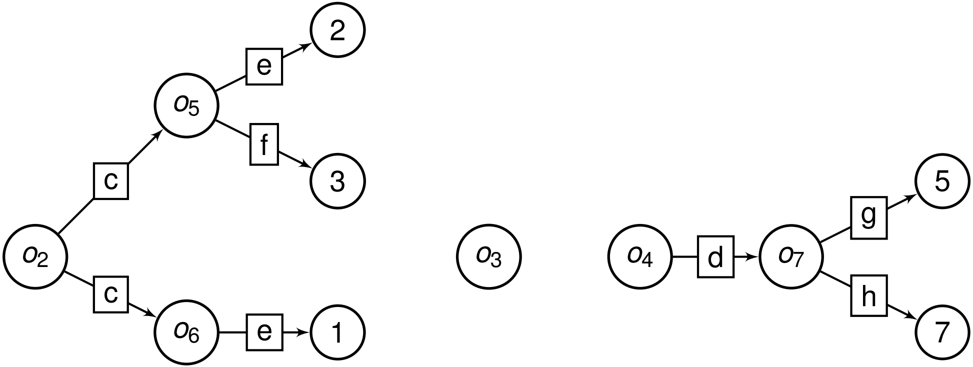
Primjer I

Neka je zadan graf G iz prethodnog primjera. Razmotrimo sljedeći upit:

$$U_5 : /\_$$

Znak _ je proizvoljan znak. Stoga su rezultat upita U_5 grafovi:

Primjer II



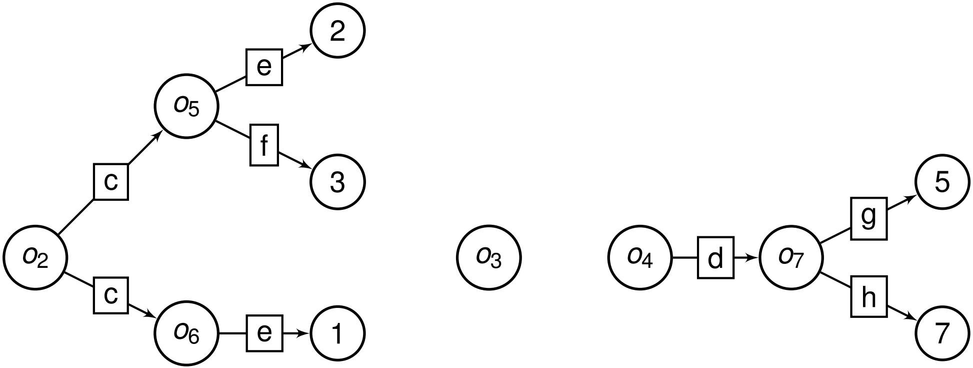
Primjer I

Neka je zadan graf G kao u prethodnom primjeru. Uvodimo alternaciju i konkatenaciju. Razmotrimo sljedeće upite:

$$U_6 : \quad /a \mid /b$$
$$U_7 : \quad /a/c$$

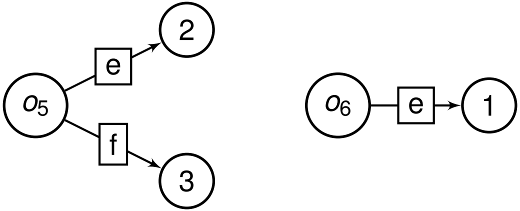
Primjer II

Znak | predstavlja "ili" stoga je rezultat upita unija rezultata upita $/a$ i $/b$, tj. grafovi:



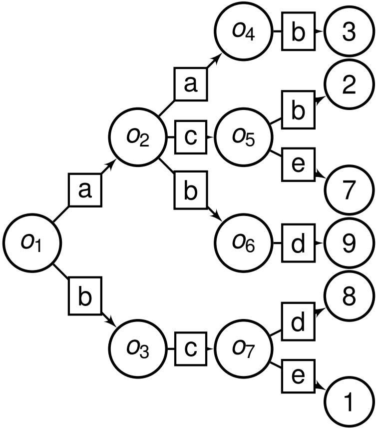
Primjer III

Spajanjem više upita u niz (konkatenacijom) radimo kompoziciju pri čemu s lijeva na desno izvršavamo upite nad rezultatima prethodnog upita. Rezultat upita U_7 su grafovi:



Primjer I

Neka je zadan graf G_1:



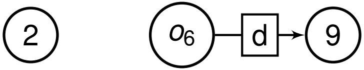
Primjer II

Razmotrimo sljedeće upite:

$$
\begin{aligned}
U_1 &: \quad /a/c?/b \\
U_2 &: \quad /a+/b \\
U_3 &: \quad /a*/b \\
U_4 &: \quad /a-/c
\end{aligned}
$$

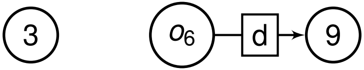
Primjer III

Znak ? označava opcionalnost. Stoga su rezultat upita U_1 grafovi:



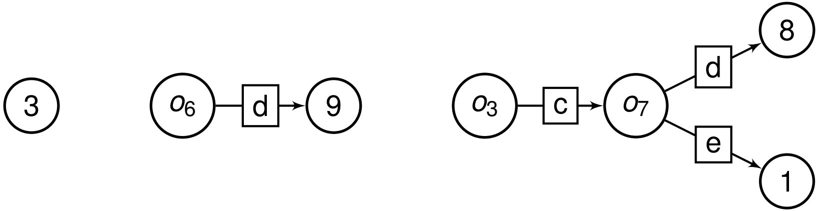
Primjer IV

Znak $+$ označava ponavljanje znaka jedan ili više puta. U skladu s time rezultat upita U_2 su grafovi:



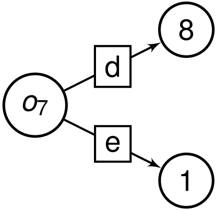
Primjer V

Znak $*$ označava ponavljanje znaka nula ili više puta. U skladu s time rezultat upita U_3 su grafovi.



Primjer VI

Znak $-$ označava negaciju, tj. na zadanom mjestu ne smije se pojaviti negirani znak. Stoga je odgovor na upit U_4:



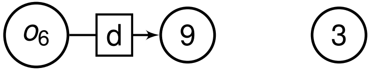
Označavanje od proizvoljnog čvora

Definicija

Označavanje počevši od proizvoljnog čvora
Posebno uvodimo oznaku $//$ koja označava označavanje od bilo kojeg čvora. Putanja oblika $//$ ... ekvivalentna je putanji $/\_*/$ ... tj. nakon korijenskog čvora slijedi niz od 0 ili više čvorova, nakon čega slijedi putanja koja se želi označiti.

Primjer

Neka je zadan podatkovni graf G_1 kao u prethodnom primjeru i upit $U : //a/b$.
Rezultat upita su sljedeći grafovi:



Proširenje $\mathbb{PIP}$ formulama

Definicija

Proširenje $\mathbb{PIP}$ formulama

Neka je zadan podatkovni graf G, nad njim definiran izraz putanje $P \in \mathbb{PIP}$. Neka je $Rez_P(G)$ rezultat upita P i neka je F formula koja je primjenjiva na $Rez[P](G)$. Rezultat upita $P[F]$ je skup podatkovnih grafova $Rez_{P[F]}(G) \subseteq Rez_P(G)$ koji zadovoljavaju formulu F.

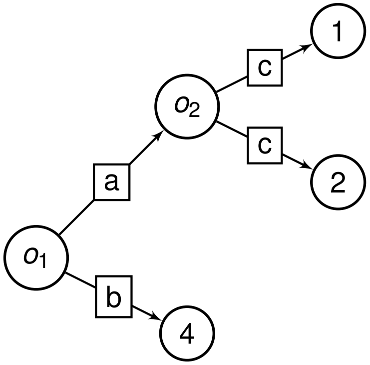
Predikati

Znakom $\cdot$ označavamo trenutni čvor (korijen razmatranog grafa). Uvodimo sljedeće predikate nad podatkovnim grafovima:

1. c_1 roditelj c_2 istinit akko je čvor c_1 roditelj čvora c_2.
2. c_1 dijete c_2 istinit akko je čvor c_1 dijete čvora c_2.
3. c_1 predak c_2 istinit akko je čvor c_1 predak čvora c_2.
4. c_1 potomak c_2 istinit akko je čvor c_1 potomak čvora c_2.
5. c_1 brat c_2 istinit akko je čvor c_1 brat čvora c_2.

Primjer I

Neka je zadan graf G:



Primjer II

U G vrijede sljedeće formule:

1. $\cdot = o_1$
2. o_1 roditelj o_2
3. 2 dijete o_2
4. o_1 predak 1
5. $\neg$(2 potomak 4)
6. 4 brat o_2
7. $\neg(o_2$ brat $o_2)$

Primjer I

Neka je zadan graf G iz prethodnog primjera:
Razmotrimo sljedeće upite:

$$
\begin{aligned}
U_1 &: \ /a/c[\cdot = 1] \\
U_2 &: \ /\_ * /c[\cdot \text{ brat } 2] \\
U_3 &: \ /a[\cdot \text{ potomak } 1 \vee \cdot \text{ potomak } 2]/c[\cdot = 2]
\end{aligned}
$$

Primjer II

Upit U_1 označava one čvorove koji se nalaze na putanji $/a/c$, a čija je vrijednost čvora jednak 1. Stoga je rezultat ovog upita graf:

Primjer III

Upit U_2 označava čvorove koji se nalaze na bilo kojoj putanji $(/\_*)$ uz uvjet da je zadnji dio putanje $/c$ i da označeni čvor ima brata čija je vrijednost 2. Rezultat ovog upita, također je graf:

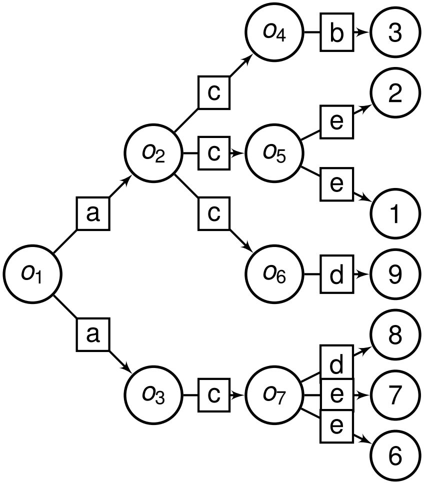
Primjer IV

Upit U_3 označava prvo čvorove koji se nalaze na putanji $/a$ i imaju potomka čija je vrijednost 1 ili 2. Nad tako dobivenim tada traži čvorove kojima odgovara putanja $/c$ i čija je vrijednost jednak 2. Dakle, rezultat upita je graf:

Primjer I

Neka je zadan podatkovni graf G:

Primjer II



Primjer III

Razmotrimo sljedeće upite:

$$
\begin{aligned}
U_1 &: /a[\exists/c/b]/c/d \\
U_2 &: /a[/c/e < 2]/c/d \\
U_3 &: /a[\forall/c/e > 5]/c/d
\end{aligned}
$$

Primjer IV

U formuli upita U_1 traži se da postoji relativna putanja u odnosu na trenutni čvor. Stoga je rezultat zadanog upita graf:

Primjer V

U formuli upita U_2 traži se da je vrijednost čvora na relativnoj putanji manja od 2. Stoga je rezultat ovog upita također graf:

Primjer VI

U formuli upita U_3 traži se da su sve vrijednosti na relativnoj putanji od trenutnog čvora veće od 5. U skladu s time je rezultat upita graf:

8

Postupak mapiranja i reduciranja

Postupak mapiranja i reduciranja (engl. map-reduce) koristi se dvama funkcijama drugog reda koje za argument primaju funkciju i skup. Prva funkcija je funkcija mapiranja, koja se definira kao što slijedi:

Definicija

Funkcija mapiranja

Neka je $f : X \to Y$ funkcija koja preslikava elemente skupa X u proizvoljan skup Y. Funkcija mapiranja $m(f, X)$ je funkcija koja će skup X preslikati u skup $Y' \subseteq Y$ tako da će na svaki element skupa X aplicirati funkciju f. Skup Y' je stoga definiran kao $\{y | y \in Y \wedge x \in X \wedge f(x) = y\}$.

Primjer

Primjer

Neka je zadan skup $X = \{1, 2, 3\}$, *skup* $Y = \mathbb{R}$ *i funkcija* $f(x) = x^2$. *Mapiranje funkcije* f *na skup* X *je tada:*

$$m(f, X) = Y' = \{1, 4, 9\}$$

Uočite da elemente skupa Y' *čine elementi skupa* X *na koje je aplicirana funkcija* f. *Također, uočite da je* $Y' \subseteq Y$.

Primjer

U nastavku ćemo funkcije mapiranja označavati u skraćenoj notaciji kao što slijedi:

$$m(f, X) = \{f(x) | \forall x \in X\}$$

Primjer

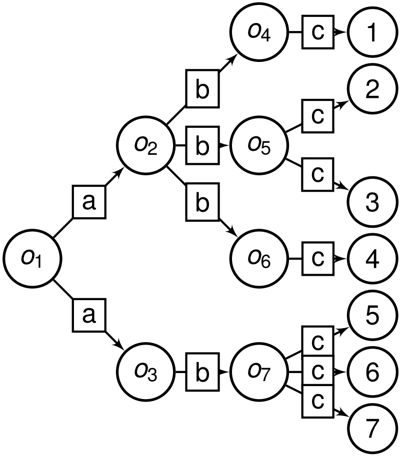
Mapiranje iz prethodnog primjera u skraćenoj notaciji izgleda ovako:

$$m(x^2, \{1, 2, 3\}) = \{x^2 | \forall x \in \{1, 2, 3\}\}$$

Funkcija mapiranja korisna je pri obradi podataka koji su izlaz iz nekog upita.

Primjer I

Neka je zadan podatkovni graf G:



Primjer II

Razmotrimo sljedeća mapiranja:

$$
\begin{aligned}
m_1 &: \{x | \forall x \in o(/a/b/c)\} \\
m_2 &: \{x \cdot 2 | \forall x \in o(//c)\} \\
m_3 &: \{o(x/c) | \forall x \in o(/a/b[c > 3])\}
\end{aligned}
$$

Primjer III

Kako bismo izračunali rezultate mapiranja, potrebno je izračunati odgovore na upite na kojima se temelje:

$$
\begin{aligned}
o(/a/b/c) &= \{1, 2, 3, 4, 5, 6, 7\} \\
o(//c) &= \{1, 2, 3, 4, 5, 6, 7\} \\
o(/a/b[c > 3]) &= \{o_6, o_7\}
\end{aligned}
$$

Primjer IV

Sada je moguće aplikacijom funkcija izračunati elemente izlaznih skupova:

$$
\begin{aligned}
m_1 &\rightarrow \{1, 2, 3, 4, 5, 6, 7\} \\
m_2 &\rightarrow \{2, 4, 6, 8, 10, 12, 14\} \\
m_3 &\rightarrow \{\{4\}, \{5, 6, 7\}\}
\end{aligned}
$$

Reduciranje

Funkcija reduciranja također je funkcija drugog reda koja se definira kao što slijedi:

Definicija

Funkcija reduciranja

Neka je $f : X, Y \to Y$ *funkcija koja prima dva argumenta iz skupa* X *i skupa* Y *te ih preslikava u elemente skupa* Y *uz uvjet da je* $X \subseteq Y$. *Funkcija reduciranja* $r(f, X)$ *je funkcija koja će preslikati skup* X *u element* $y \in Y$ *tako da će skup* X *reducirati putem funkcije* f, *tj. ako je* $X = \{x_1, x_2, x_3, \ldots, x_n\}$ *tada će rezultat redukcije biti* $y = f(\ldots f(f(x_1, x_2), x_3) \ldots, x_n)$ *odnosno ako je* $X = \{x\}$ *(skup ima samo jedan element) tada je* $y = x$.

Primjer

Primjer

Neka je zadan skup $X = \{1, 2, 3\}$*, skup* $Y = \mathbb{R}$ *i funkcija* $f(x, y) = x \cdot y$*, tada je redukcija skupa* X *putem funkcije* f*:*

$$r(f, X) = y = f(f(1, 2), 3) = f(1 \cdot 2, 3) = f(2, 3) = 2 \cdot 3 = 6$$

Redukcije nam omogućavaju da rezultate iz upita (ili mapiranja) reduciramo, slično agregirajućim funkcijama u SQL-u.

Primjer I

Neka je zadan graf G kao u prethodnom primjeru te funkcija $f(x, y) = x + y$. Izračunajmo sljedeće redukcije:

$$r_1(f, o(//c))$$
$$r_2(f, o(//b[c < 3]/c))$$
$$r_3(f, \{z^2 | \forall z \in o(//c[.\%2 = 1])\})$$

Primjer II

Izračunajmo prvo odgovore na upite:

$$
\begin{aligned}
o(//c) &= \{1,2,3,4,5,6,7\} \\
o(//b[c<3]/c) &= \{1,2\} \\
o(//c[.\%2=1]) &= \{1,3,5,7\}
\end{aligned}
$$

Posebno za treću redukciju, moramo izračunati mapiranje:

$$\{z^2 | \forall z \in o(//c[.\%2=1])\} = \{1,9,25,49\}$$

Primjer III

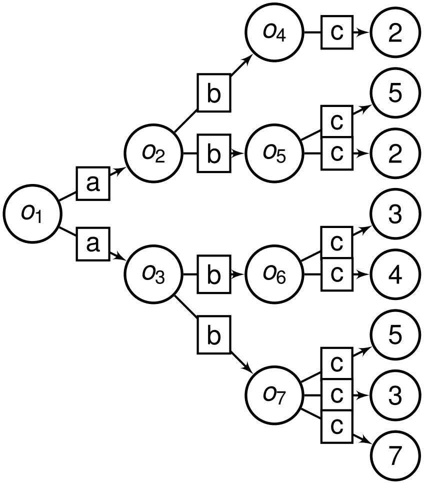
Sada je moguće izračunati redukcije:

$$
\begin{aligned}
r_1 &\longrightarrow 28 \\
r_2 &\longrightarrow 3 \\
r_3 &\longrightarrow 84
\end{aligned}
$$

Primjer I

Neka je zadan podatkovni graf G:

Primjer II



Primjer III

Neka je zadan upit U odrediti sume čvorova c grupirano prema čvorovima b čija su djeca.

Neka je zadana funkcija $f(x, y) = x + y$. Sljedeći izraz daje rješenje zadatka:

$$m : \{r(f, o(x/c)) | x \in o(//b)\}$$

Izračunajmo rezultat upita $//b$:

$$o(//b) = \{o_4, o_5, o_6, o_7\}$$

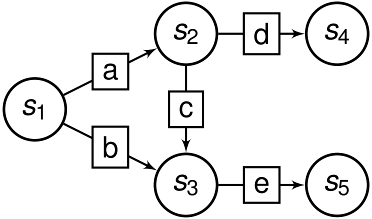
Primjer IV

stoga je mapiranje:

$$
\begin{aligned}
& \{r(f, o(x/c)) | x \in \{o_4, o_5, o_6, o_7\}\} \\
= \ & \{r(f, o(o_4/c)), r(f, o(o_5/c)), r(f, o(o_6/c)), r(f, o(o_7/c))\} \\
= \ & \{r(f, \{2\}), r(f, \{5, 3\}), r(f, \{3, 4\}), r(f, \{5, 3, 7\})\} \\
= \ & \{2, 8, 7, 15\}
\end{aligned}
$$

Zadaci I

Neka je zadan graf sheme S:



Zadaci II

I sljedeća ograničenja:

$$
\begin{aligned}
Dom(s_4) &= \{\alpha, \beta, \gamma, \delta\} \\
Dom(s_5) &= \{1, 2, 3\} \\
Kard_{min}(a) &= 1 \\
Kard_{max}(a) &= 2 \\
Kard_{min}(c) &= 2
\end{aligned}
$$

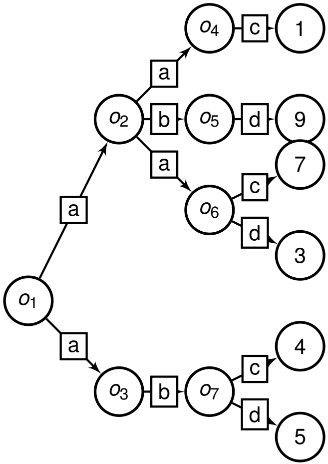
a) Napišite G koji je valjana instanca grafa sheme S.
b) Provjerite koristeći opisani algoritam je li S minimalni graf sheme za G.

Zadaci III

c) Je li postoji graf G' za koji vrijedi da je instanca grafa sheme S i da ima minimalni graf sheme različit od S? Zašto je to tako?

Neka je zadan podatkovni graf G:

Zadaci IV



Zadaci V

Izračunajte rezultate sljedećih upita:

a) $/a * /b * /c$
b) $/a * /b + /c$
c) $/a/a \mid /a/b$
d) $/a + /b?/d$
e) $/\_ * /a - /c$
f) $/a/\_[\cdot \text{ roditelj } /c \wedge \cdot \text{ roditelj } /d]$
g) $/\_[\exists/b/c]/b$
h) $//a$
i) $//\_$

Pitanja?

Izvor

Maleković, M., Schatten, M. (2017) Teorija i primjena baza podataka, Fakultet organizacije i informatike, Varaždin.