

რელატივის ალგორითმი მონაცემთა ბაზის მუშის რეალური სისრულის უზრუნველსაყოფად

ა. კობიაშვილი, ე. გვარამია
საქართველოს ტექნიკური უნივერსიტეტი

რეზიუმე

შემოთავაზებულია ალგორითმი, რომელიც უზრუნველყოფს რელაციური აღრიცხვის ნებისმიერი გამოსახულების დაყვანას რელაციური ალგებრის სემანტიკურად ექვივალენტურ გამოსახულებამდე. წარმოდგენილია მოცემული ალგორითმის საილუსტრაციო მაგალითი. ნაჩვენებია ამ ალგორითმის უპირატესობა.

საკვანძო სიტყვები: მონაცემთა ბაზა, რელაციური აღრიცხვა, რელაციური ალგებრა, რელაციური სისრულე, კვანტორი.

1. შესავალი

რელაციური მოდელის ერთ-ერთი უმნიშვნელოვანესი ნაწილის – მანიპულაციური ნაწილის – მნიშვნელოვანი კომპონენტებია რელაციური აღრიცხვა და რელაციური ალგებრა. ისინი ფუნდამენტურად ეკვივალენტურია [1]. ერთადერთი განსხვავება იმაში მდგომარეობს, რომ რელაციური ალგებრა მოიცავს რელაციური დამოკიდებულების აგებისათვის აუცილებელ ოპერაციებს, ხოლო რელაციური აღრიცხვა უბრალოდ განსაზღვრავს სასურველ დამოკიდებულებებს.

2. ძირითადი ნაწილი

იმისათვის, რომ რელაციური აღრიცხვა გახდეს რელაციური ალგებრის სავსებით ეკვივალენტური, საჭიროა შეიქმნას ალგორითმი, რომლის მიხედვითაც რელაციური აღრიცხვის ნებისმიერი გამოსახულება შეიძლება დაყვანილ იქნას მის სემანტიკურად ეკვივალენტურ ალგებრულ გამოსახულებამდე. აღნიშნული ალგორითმი განვიხილოთ კონკრეტული მაგალითის მეშვეობით.

S	S#	SNAME	STATUS	CITY	SPJ	S#	P#	J#	QTY
	S1	Smith	20	London		S1	P1	J1	200
	S2	Jones	10	Paris		S1	P1	J4	700
	S3	Blake	30	Paris		S2	P3	J1	400
	S4	Clark	20	London		S2	P3	J2	200
	S5	Adams	30	Athens		S2	P3	J3	200
						S2	P3	J4	500
						S2	P3	J5	600
P	P#	PNAME	COLOR	WEIGHT	CITY	S2	P3	J6	400
	P1	Nut	Red	12	London	S2	P1	J7	800
	P2	Bolt	Green	17	Paris	S2	P5	J2	100
	P3	Screw	Blue	17	Rome	S3	P3	J1	200
	P4	Screw	Red	14	London	S3	P4	J2	500
	P5	Cam	Blue	12	Paris	S4	P6	J3	300
	P6	Cog	Red	19	London	S4	P6	J7	300
						S5	P2	J2	200
						S5	P2	J4	100
						S5	P5	J5	500
J	J#	JNAME		CITY		S5	P5	J7	100
	J1	Sorter		Paris		S5	P6	J2	200
	J2	Punch		Rome		S5	P1	J4	100
	J3	Reader		Athens		S5	P3	J4	200
						S5	P4	J4	800
	J4	Console		Athens		S5	P5	J4	400
	J5	Collater		London		S5	P6	J4	500
	J6	Terminal		Oslo					
	J7	Tape		London					

მოცემულ მაგალითს საფუძვლად უდევს მარტივი მონაცემთა ბაზა, რომელიც შედგება სამი ცხრილისაგან: S ცხრილი, რომელშიც ჩაწერილია ინფორმაცია საწარმოს მიწოდებლების შესახებ, P ცხრილი, რომელიც საწარმოს მიერ გამოყენებული ნაწილების შესახებ შეიცავს ინფორმაცია, SPJ ცხრილი, რომელიც გვიჩვენებს დამოკიდებულებებს საწარმოს მიწოდებლებსა და მიწოდებულ ნაწილებს შორის და J ცხრილი, რომელშიც ინახება ინფორმაცია პროექტების შესახებ. S ცხრილი პირველადი გასაღების S# სვეტი, მეორისა – P# სვეტი, მესამე – SPY ცხრილისა – კომბინაცია S#/P#, ხოლო J ცხრილისა – J# სვეტი.

ახლა განვიხილოთ მოთხოვნა: „გამოვიტანოთ იმ მიწოდებლების სახელები და ქალაქის სახელები, რომლების სულ ცოტა ერთ ათენურ პროექტს ამარაგებს ნაწილებით სულ ცოტა 51 ცალით თითოეული სახეობისათვის“. რელაციურ აღრიცხვაში ამ მოთხოვნას ექნება შემდეგი სახე:

```
SX.SNAME, SX.CITY WHERE EXISTS JX FORALL PX
EXISTS SPJX
(JX . CITY='Athens' AND
JX . J# = SPJX . J# AND
PX . P# = SPJX . P# AND
SX . S# = SPJX . S# AND
SPYX . QTY >50)
```

სადაც SX, PX, JX, SPJX წარმოადგენს კორტეჟების ცვლადებს შესაბამისად S, P, J, SPJ დიაპაზონიდან.

ბიჯი 1. კორტეჟის ყველა ცვლადისათვის ვიპოვოთ შეზღუდვათა დიაპაზონი. ამისათვის WHERE ფრაზაში უნდა ჩავრთოთ შეზღუდვის პრედიკატი, რომელიც განსაზღვრულ კორტეჟებს გამორიცხავს შემდგომი განხილვიდან. კერძოდ, მოცემულ შემთხვევაში გამოირიცხება შემდეგი კორტეჟები:

SX: S-ის ყველა კორტეჟი 5 კორტეჟი
 PX: P-ს ყველა კორტეჟი 5 კორტეჟი
 JX: J-ის ის კორტეჟები, სადაც CITY = 'Athens' 2 კორტეჟი
 SPJX: SPJ-ის კორტეჟები, სადაც QTY = 'Athens' 24 კორტეჟი

ბიჯი 2. შევადგინოთ წინა ბიჯზე მიღებული ყველა დიაპაზონის დეკარტული ნამრავლი:

S#	SN	STATUS	CITY	P#	PN	COLOR	WEIGHT	CITY	J#	JN	CITY	S#	P#	J#	STY
S1	Sm	20	Lon	P1	Nt	Red	12	Lon	J3	Rd	Ath	S1	P1	J1	200
S1	Sm	20	Lon	P1	Nt	Red	12	Lon	J3	Rd	Ath	S1	P1	J4	700
.
.

სრული ნამრავლი შეიცავს $5*6*2*24=1440$ კორტეჟს.

ბიჯი 3. შევზღუდოთ დეკარტის ნამრავლი ისე, რომ ავაგოთ WHERE ფრაზის შეერთების პირობით გათვალისწინებული შედეგი. ეს პირობაა:

JX . J# = SPJX . J# AND PX . P# = SPJX . P# AND SX . S# = SPJX . S#

ამით გამოვრიცხვათ ნამრავლიდან იმ კორტეჟებს რომელთათვისაც S# მიწოდებლის სიდიდე ტოლი არაა SP# მიწოდებლის ან P# ნაწილის სიდიდე ტოლი არაა SP# მიწოდების სიდიდის ან J# პროექტის ნომერი ტოლი არაა J# მიწოდების სიდიდის. ამჯერად მიიღება დეკარტის ნამრავლი, შედეგნილი უკვე მხოლოდ 10 კორტეჟისაგან.

S#	SN	STATUS	CITY	P#	PN	COLOR	WEIGHT	CITY	J#	JN	CITY	S#	P#	J#	QTY
S1	Sm	20	Lon	P1	Nt	Red	12	Lon	J4	Cn	Ath	S1	P1	J4	700
S2	Jo	10	Par	P3	Sc	Blue	17	Rom	J3	Rd	Ath	S2	P3	J3	200
S2	Jo	10	Par	P3	Sc	Blue	17	Rom	J4	Cn	Ath	S2	P3	J4	500
S4	Cl	20	Lon	P6	Cg	Red	19	Lon	J3	Rd	Ath	S4	P6	J3	300
S5	Ad	30	Ath	P2	Bt	Green	17	Par	J4	Cn	Ath	S5	P2	J4	100
S5	Ad	30	Ath	P1	Nt	Red	12	Lon	J4	Cn	Ath	S5	P1	J4	100
S5	Ad	30	Ath	P3	Sc	Blue	17	Rom	J4	Cn	Ath	S5	P3	J4	200
S5	Ad	30	Ath	P4	Sc	Red	14	Lon	J4	Cn	Ath	S5	P4	J4	800
S5	Ad	30	Ath	P5	Cm	Blue	12	Par	J4	Cn	Ath	S5	P5	J4	400
S5	Ad	30	Ath	P6	Cg	Red	19	Lon	J4	Cn	Ath	S5	P6	J4	500

ბიჯი 4. გამოვიყენოთ კვანტორები მარჯვნიდან მარცხნივ შემდეგნაირად:

- “Exists Rx“ კვანტორისათვის განვსაზღვროთ მიმდინარე შუალედური შედეგის პროექცია R დამოკიდებულების ყველა კორტეჟის გამოსარიცხად.

- “Forall RX“ კვანტორისათვის მიმდინარე შუალედური სედეგი გავყოთ პირველ ბიჯზე მიღებული დამოკიდებულების შეზღუდვის დიაპაზონზე.

- მოცემულ მაგალითში კვანტორებია:

EXISTS JX FORALL PX EXISTS SPJX

აქედან გამომდინარე:

1. SPJ . S#, SPJ . P#, SPJ . J#, SPJ . QTY ატრიბუტების პროექციას ექნება სახე:

S#	SN	STATUS	CITY	P#	PN	COLOR	WEIGHT	CITY	J#	JN	CITY	J#	JN	J#
S1	Sm	20	Lon	P1	Nt	Red	12	Lon	J4	Cn	Ath	J4	Cn	J4
S2	Jo	10	Par	P3	Sc	Blue	17	Rom	J3	Rd	Ath	J3	Rd	J3
S2	Jo	10	Par	P3	Sc	Blue	17	Rom	J4	Cn	Ath	J4	Cn	J4
S4	Cl	20	Lon	P6	Cg	Red	19	Lon	J3	Rd	Ath	J3	Rd	J3
S5	Ad	30	Ath	P2	Bt	Green	17	Par	J4	Cn	Ath	J4	Cn	J4
S5	Ad	30	Ath	P1	Nt	Red	12	Lon	J4	Cn	Ath	J4	Cn	J4
S5	Ad	30	Ath	P3	Sc	Blue	17	Rom	J4	Cn	Ath	J4	Cn	J4
S5	Ad	30	Ath	P4	Sc	Red	14	Lon	J4	Cn	Ath	J4	Cn	J4
S5	Ad	30	Ath	P5	Cm	Blue	12	Par	J4	Cn	Ath	J4	Cn	J4
S5	Ad	30	Ath	P6	Cg	Red	19	Lon	J4	Cn	Ath	J4	Cn	J4

2. P დამოკიდებულებაზე გაყოფას ექნება შემდეგი სახე:

S#	SN	STATUS	CITY	J#	JN	CITY
S5	Ad	30	Ath	J4	Cn	ATH

ახლა ეს უკანასკნელი დამოკიდებულება ვაჩვენოთ სრულად:

S#	SNAME	STATUS	CITY	J#	JNAME	CITY
S5	Adams	30	Athens	J4	Console	Athens

3. მოვასწავლოთ J#, JNAME, CITY ატრიბუტების პროექცია:

S#	SNAME	STATUS	CITY
S5	Adams	30	Athens

ბიჯი 5. განვიხილოთ მეოთხე ბიჯის შედეგების პროექცია მიზნობრივი სიის სპეციფიკაციების შესაბამისად. მიზნობრივ სიას აქვს სახე:

SX . SNAME, SX . CITY

ამრიგად საბოლოო შედეგს ექნება შემდეგი სახე:

SNAME	CITY
Adams	Athens

3. დასკვნა

აღნიშნული ალგორითმი უზრუნველყოფს მონაცემთა ბაზების ნებისმიერი ენის რელაციურ სისრულეს. რელაციური სისრულის ქვეშ იგულისხმება მონაცემთა ბაზების ენებში მონაცემთა ამორჩევისა და მათი გამოსახვის საშუალებათა შესაძლო სიმძლავრე.

ამასთან, რადგანაც ალგებრა რელაციურად სრულია, იმის საჩვენებლად, რომ ნებისმიერი მოცემული L ენაც სრულია, საჭიროა ვაჩვენოთ, რომ L მოიცავს ყოველი ალგებრული ოპერაციის ანალოგს. რელაციურად სრული ენის მაგალითებია QUEL და SQL.

მოცემული ალგორითმით ჩვენ ვაჩვენებთ, რომ აღრიცხვის ნებისმიერი გამოსახულება შეიძლება დაყვანილ იქნას მის ალგებრულ ეკვივალენტზე და აქედან გამომდინარე, აღრიცხვა ისეთივე მძლავრი ხდება როგორც აღრიცხვა.

4. ლიტერატურა

1. Codd E.F. Relational Completeness of Data Base Sublanguages. In Data Base Systems: Courant Computer Science Symposia Series, Vol.6. Englewood Cliffs, N.J. : PrenticeHall, 1972.
2. Pirotte A., Wodon P. A comprehensive Formal Query Language for a Relational Data Base. R.A.I. R.O. Informatique / Computer Science 11, No. 2, 1997.

АЛГОРИТМ РЕДУКЦИИ ДЛЯ ОБЕСПЕЧЕНИЯ РЕЛЯЦИОННОЙ ПОЛНОТЫ ЯЗЫКОВ БАЗ ДАННЫХ.

Кобиашвили А., Гварамиа Э.
Грузинский Технический Университет

Резюме

Предложен алгоритм, позволяющий произвольное выражение реляционного исчисления, приводить к семантически эквивалентному выражению реляционной алгебры. Представлен пример, иллюстрирующий данный алгоритм. Показано преимущество данного алгоритма.

A REDUCTION ALGORITHM GUARANTEEING RELATIONAL COMPLETENESS OF DATABASE LANGUAGES.

Kobiashvili A., Gvaramia E.
Georgian Technical University

Summary

A reduction algorithm by which an arbitrary expression of the relational calculus can be reduced to a semantically equivalent expression of the relational algebra is offered. The example illustrating this algorithm is presented. The advantage of this algorithm is shown.