Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Predmet:

Osnove PB

Modul:

Relacijski račun

Gradivo:

v.2019

Relacijski poizvedovalni jeziki

- Relacijska algebra in relacijski račun formalna poizvedovalna jezika...
  - Relacijska algebra visoko-nivojski postopkovni jezik,
  - Relacijski račun nepostopkovni ali deklarativni jezik.

- Formalno ekvivalentna.

- Relacijsko popolni jeziki

Deklarativno poizvedovanje

- Relacijski račun je deklarativni poizvedovalni jezik.
  - V nasprotju z relacijsko algebro, kjer (implicitno) določimo postopek, s katerim pridemo do rezultata, pri relacijskem računu določimo le, kaj nas zanima.

- Temelji na simbolični logiki imenovani predikatni račun prvega reda.
  - V naravni jeziku so stavki sestavljeni iz besed, ki sledijo sintaksi. Besede, ki označujejo trditve, imenujemo *predikati* in besede o katerih trditve držijo, imenujemo *konstante*.
    - Objekt a ima lastnost P.
    - Objekta a_1, a_2 sta v odnosu Q.

Stavki v predikatnem računu

- Stavki v predikatnem računu so sestavljeni iz predikatov in konstant. Predikat je logična funkcija z argumenti.

- Primer:
  - `Marija je ženska.`
  - `Janez je moški.`
  - `Marija in Janez sta sestra in brat.`

predikat   konstanta

  - `Ženska(marija)`
  - `Moški(janez)`
  - `SestraBrat(marija,janez)`

Spremenljivke in kvantifikatorji

- Predikati imajo lahko spremenljivke kot argumente, katerih vrednosti so lahko kvantificirane.

- Primer:
  - Vsakdo je bodisi moški ali ženska.
  - Moški ni ženska.

$$\forall x.\ \text{Moški}(x) \lor \text{Ženska}(x)$$

$$\boxed{\forall} x.\ \text{Moški}(x) \Rightarrow \neg\text{Ženska}(\boxed{x})$$

kvantifikator

spremenljivka

Funkcije

- V predikatnem računu so domene objektov lahko določene s funkcijami, ki so aplicirane na objekte.

- Primer:
  - Oče osebe je moški

$$\forall x.\ \text{Moški}(\underbrace{\boxed{\text{Oče}}}_{\text{Funkcija}}(x))$$

Sintaksa predikatnega računa

- Če predikat vsebuje spremenljivko (npr. `x je študent`), mora za x obstajati domena vrednosti.
  - Za nekatere vrednosti iz domene je trditev resnična za druge neresnična.

- Sintaksa: če je P predikat, potem lahko zapišemo množico vseh x, za katere je P resničen, takole:

  $\{x \mid P(x)\}$

- Predikate lahko povezujemo z logičnimi operatorji: IN ($\wedge$), ALI ($\vee$), NEGACIJA ($\neg$)

Vrste relacijskega računa

- V povezavi s podatkovnimi bazami poznamo dve vrsti relacijskega računa:
  - N-terični relacijski račun (*Tuple Relational Calculus*)
    - Spremenljivke se nanašajo na n-terice ali vrstice relacije.
  - Domenski relacijski račun (*Domain Relational Calculus*).
    - Spremenljivke se nanašajo na domene atributov.

N-terični relacijski račun

- Temelji na uporabi n-teričnih spremenljivk.
  - N-terična spremenljivka je spremenljivka, katere domena je določena z relacijo, t.j. spremenljivka, katere dovoljene vrednosti so n-terice relacije.

- Splošna oblika zapisa n-teričnega računa je:

```
{t | COND(t)}
```

  - kjer `t` prestavlja n-terično spremenljivko, `COND(t)` pa nabor pogojev, ki naj jih n-terice, ki jih iščemo, izpolnjujejo.

Primeri

- Poišči vse podatke o artiklih, ki imajo kritično zalogo manjšo od tri (zaloga < 3):

Označba domene! Za iskane n-terice zahtevamo, da so iz relacije Artikel.

$$\{A \mid \boxed{Artikel(A)} \wedge A.zaloga < 3\}$$

- Če nas zanima samo določen atribut (npr. naziv artikla), zapišemo:

$$\{A.naziv \mid Artikel(A) \wedge A.zaloga < 3\}$$

Uporaba kvantifikatorjev

- Na koliko primerkov se predikat nanaša, določimo s pomočjo kvantifikatorjev.
- Obstajata dva kvantifikatorja:
  - Eksistencialni kvantifikator $\exists$ (beremo 'obstaja')
  - Univerzalni kvantifikator $\forall$ (beremo 'za vse')

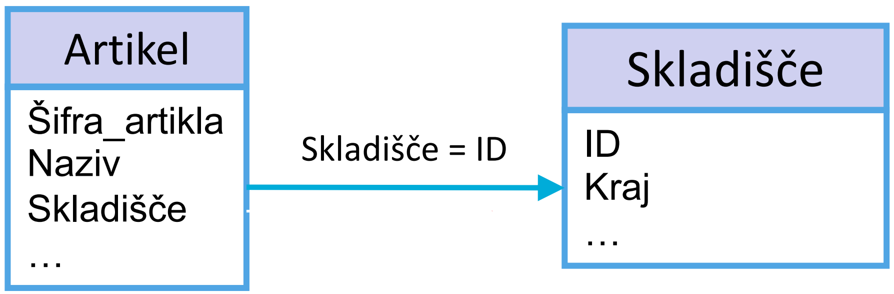
- N-terične spremenljivke, ki so kvantificirane s kvantifikatorjema $\forall$ ali $\exists$, imenujemo vezane spremenljivke, ostale pa proste spremenljivke.

Eksistencialni kvantifikator

- Eksistencialni kvantifikator uporabimo v izrazih, ko želimo povedati, da mora obstajati vsaj en primerek, za katerega je predikat resničen.

- Primer: iščemo zapise iz relacije `Artikel`, ki se nahajajo v skladišču v Ljubljani:

$$\{A \mid \text{Artikel}(A) \land (\exists S)(\text{Skladišče}(S) \land (A.\text{skladišče} = S.\text{ID}) \land (S.\text{kraj} = \text{‘Ljubljana’}))\}$$

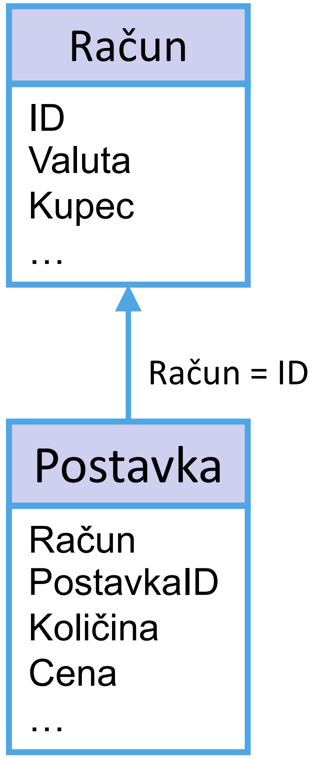


Univerzalni kvantifikator

- Univerzalni kvantifikator uporabimo v izrazih, ki se nanašajo na vse primerke…
- Primer: iščemo račune, ki imajo ceno vsake postavke večjo od 1.000 EUR

$$\{R \mid \text{Račun}(R) \land (\forall P)(\text{Postavka}(P) \land (R.ID = P.\text{račun}) \land (P.\text{cena} > 1000))\}$$

Ali lahko isti rezultat pridobimo z uporabo negacije?



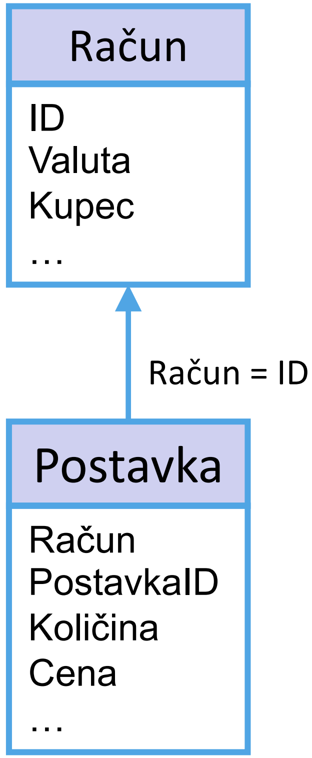
Univerzalni kvantifikator

- Univerzalni kvantifikator uporabimo v izrazih, ki se nanašajo na vse primerke…
- Primer: iščemo račune, ki imajo ceno vsake postavke večjo od 1.000 EUR

$$\{R \mid \text{Račun}(R) \land (\forall P)((\text{Postavka}(P) \land (R.ID = P.\text{račun})) \Rightarrow (P.\text{cena} > 1000))\}$$

$$\{R \mid \text{Račun}(R) \land (\forall P)(\neg(\text{Postavka}(P) \land (R.ID = P.\text{račun})) \lor (P.\text{cena} > 1000))\}$$

$$\{R \mid \text{Račun}(R) \land \neg(\exists P)(\text{Postavka}(P) \land (R.ID = P.\text{račun}) \land (P.\text{cena} <= 1000)\}$$



Proste in vezane spremenljivke

- Proste spremenljivke v izrazih relacijskega računa so lahko samo spremenljivke, ki so definirane oz. se nahajajo na levi strani znaka | v izrazu.

- Primer:

  {A.naziv, A.cena | Artikel(A) ∧ (∃S)(Skladišče(S) ∧ (A.skladišče = S.ID) ∧ S.kraj = 'Ljubljana')}

  - A je prosta spremenljivka, S pa vezana!

Dobri izrazi

- Izrazi v relacijskem računu morajo biti dobri, t.j. nedvoumni in smiselni.
- Splošna oblika dobrega izraza je naslednja:

  `{S1.a1, S2.a2,..., Sn.an | F(S1, S2,..., Sm)}   m≥n`

- kjer je pomen naslednji:
  - `S1, S2,..., Sm` so n-terične spremenljivke
  - `Sj.ai` so atributi relacije, ki je domena n-terične spremenljivke `Si`
  - `F` je formula.

Dobro definirana formula

- Dobro definirana formula je sestavljena iz naslednjih atomov:
  - `R(Si)`, kjer je `Si` n-terična spremenljivka in `R` relacija.
  - `Si.a1` θ `Sj.a2`, kjer sta `Si` in `Sj` n-terični spremenljivki, `a1` atributi relacije, ki je domena `Si`, `a2` atributi relacije, ki je domena `Sj`, θ pa primerjalni operator ($<$, $\leq$, $>$, $\geq$, $=$, $\neq$).
  - `Si.a1` θ `c`, kjer je `Si` n-terična spremenljivka, `a1` atributi relacije, ki je domena `Si`, θ primerjalni operator, `c` pa konstanta iz domene atributov `a1`.

Pravila gradnje formul

- Formule gradimo rekurzivno iz atomov.

- Upoštevamo naslednja pravila:
  - Atom je že sam formula
  - Če sta `F1` in `F2` formuli, so formule tudi konjunkcija `F1∧F2`, disjunkcija `F1∨F2`, in negacija `¬F1`
  - Če je `F` formula s prosto spremenljivko `X`, potem sta `(∃X)(F)` in `(∀X)(F)` tudi formuli.

Primeri

- Imamo naslednje relacije:

```
Hotel   (hotelNo, hotelName, address)
Room    (roomNo, #hotelNo, type, price, free)
Booking (#hotelNo, #guestNo, dateFrom, dateTo, #roomNo)
Guest   (guestNo, guestName, guestAddress)
```

- Izpiši nazive hotelov, ki se nahajajo v Ljubljani.

- Izpiši nazive hotelov, ki imajo prosto vsaj eno dvoposteljno sobo (type=2).

Primeri

- Izpiši nazive hotelov, ki so trenutno brez gostov.

Varni izrazi

- V relacijskem računu je možno zapisati stavke, ki vračajo neskončne množice.
- Varnost izrazov dosežemo z omejitvijo, da morajo biti vse vrednosti, ki se pojavijo v rezultatu, vrednosti iz domene izraza `E` `(dom(E))`.
  - `dom(E)`: vrednosti, ki se eksplicitno pojavijo v izrazu `E` kot konstante ali pa kot zapis ene od relacij, ki nastopajo v `E`.

- Primer nevarnega izraza:

  `{A | ¬(Artikel(A)) }`

Sintaksa domenskega računa

- Splošna oblika izraza v domenskem relacijskem računu je:

  `{d1, d2,..., dn | F(d1, d2,..., dm)}`

- kjer je:

  `d1, d2,..., dm` množica domenskih spremenljivk in

  `F(d1, d2,..., dm)` formula.

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Formule

- Formula `F` je sestavljena iz atomov oblike:
  - `R(d1, d2,..., dn)`, kjer je `R` relacija, di pa domenske spremenljivke
  - `di θ dj`, kjer sta di in dj domenski spremenljivki, θ pa primerjalni operator ($<$, $\leq$, $>$, $\geq$, $=$, $\neq$)
  - `di θ c`, kjer je `di` domenska spremenljivka, θ primerjalni operator, `c` pa konstanta iz domene `di`

Gradnja izrazov

- Formule gradimo rekurzivno iz atomov.

- Upoštevamo naslednja pravila:
  - Atom je že sam formula
  - Če sta F1 in F2 formuli, so formule tudi konjunkcija F1∧F2, disjunkcija F1∨F2, in negacija ¬F1
  - Če je F formula z nevezano domensko spremenljivko X, potem sta (∃X)(F) in (∀X)(F) tudi formuli.

Primeri

- Imamo naslednje relacije:

```
Hotel   (hotelNo, hotelName, address)
Room    (roomNo, #hotelNo, type, price, free)
Booking (#hotelNo, #guestNo, dateFrom, dateTo, #roomNo)
Guest   (guestNo, guestName, guestAddress)
```

- Izpiši imena hotelov v Ljubljani.

Primeri

- Izpiši nazive hotelov, ki imajo prosto vsaj eno dvoposteljno sobo (type=2)

N-terični in domenski relacijski račun

- N-terični relacijski račun *(Edgar F. Codd, 1972)*:
  - uporabljamo spremenljivke, katerih zaloga vrednosti so relacije.
  - SQL temelji na n-teričnem relacijskem računu!
  - IBM (raziskovalni laboratorij v San Jose, California)

- Domenski relacijski račun *(M. Lacroix in A. Pirotte, 1977)*:
  - uporabljamo spremenljivke, katerih zaloga vrednosti so domene atributov.
  - QBE temelji na domenskem relacijskem računu!
  - IBM (raziskovalni laboratorij v Yorktown Heights, New York).

Primerjava jezikov

- N-terični relacijski račun

  {H.hotelName | Hotel(H) ∧ (∃R)(Room(R) ∧ H.hotelNo=R.hotelNo ∧ R.type=2 ∧ R.free = true)}

- Domenski relacijski račun

  {hName | ∃hNo, hName, hAddress, rNo, rPrice: (Hotel (hNo, hName, hAddress) ∧ (Room(rNo, hNo, 2, rPrice, true)) }

- Relacijska algebra

  $$\pi_{H.hotelName}(\sigma_{R.type=2\ \wedge\ R.free=true}(H \bowtie_{H.hotelNo=R.hotelNo} R))$$

- SQL

```
SELECT h.hotelName FROM Hotel h, Room r
WHERE
    h.hotelNo=r.hotelNo AND r.type=2 AND r.free=true
```

Univerza *v Ljubljani*
Fakulteta *za računalništvo in informatiko*

Moč formalnih jezikov

- Ob uporabi varnih izrazov so si relacijska algebra, n-terični relacijski račun in domenski relacijski račun po moči enakovredni:
  - Kar lahko izrazimo v relacijski algebri, lahko izrazimo tudi v n-teričnem ali domenskem relacijskem računu.
  - Vsak varen izraz v relacijskem računu lahko zapišemo tudi z relacijsko algebro.

- Vsak jezik, s katerim lahko pridobimo relacije, ki jih je moč pridobiti z relacijskim računom, je relacijsko popoln (*relationally complete*).