

1.4. OSNOVNI POJMOVI LOGIČKIH STRUKTURA PODATAKA

U ovoj tački su izložene definicije, ilustrovane primerima, jednog broja osnovnih pojmoveva, vezanih za logičke strukture nad skupom obeležja i logičke strukture nad skupom podataka. U tom smislu, definisani su pojmovi: obeležja, podatka, tipa i pojave entiteta, odnosno sloga, datoteke i ključa.

→ OBELEŽJE I PODATAK

Zadatak automatizovanih informacionih sistema predstavlja prikupljanje, memorisanje, obrada, prenos i prezentiranje podataka o entitetima^{*)} raznih klasa nekog realnog sistema. Skup sličnih entiteta naziva se **klasom**. Klase entiteta predstavljaju: klase subjekata, kao što su

^{*)} Ako se prihvati da petlje u strukturama podataka nemaju smisla, tada do $N - 1$.

^{*)} Engleska reč entity se, po pravilu, ne prevodi na srpski jezik. Približno značenje joj je jedinica posmatranja

radnici jedne radne organizacije ili studenti nekog univerziteta, klase objekata, kao što su proizvodi, zgrade ili organizacije, klase događaja, kao što su uplata na štednu knjižicu ili upis u školu, klase raznih pojmoveva i pojava. Treba zapaziti da kriterijum sličnosti igra značajnu ulogu u definisanju klase entiteta. U zavisnosti od usvojenog kriterijuma sličnosti, dva entiteta se mogu naći u istoj ili u dve različite klase. Isto tako, isti entitet može pripadati različitim klasama.

Primer 1.14. Studenti različitih fakulteta istog univerziteta mogu pripadati klasama entiteta studenata odgovarajućih fakulteta ili klasi entiteta studenata univerziteta. □ ✓

Svi entiteti jedne klase poseduju odredene zajedničke osobine, kao što su: naziv, boja, vrednost, trajanje i slično. Ove osobine nazivaju se obeležjima (atributima). Obeležja se označavaju velikim slovima latinske azbuke, skraćenim nazivom (mnemonikom) ili punim nazivom. Na primer, A , X , MBR ili $MATIČNI_BROJ_RADNIKA$.

Svakom od obeležja odgovara jedan skup svih mogućih vrednosti koje to obeležje, u konkretnim slučajevima, može imati. Taj skup vrednosti se naziva domenom obeležja. Domen obeležja A se obeležava sa $\text{dom}(A)$. Domen se naziva i tipom podataka.

Primer 1.15. Za obeležje $BOJA_AUTOMOBILA$ skup vrednosti je

$$\text{dom}(BOJA_AUTOMOBILA) = \{\text{bela}, \text{žuta}, \text{crna}, \text{plava}, \dots\}. \square$$

U strukturama podataka, pojam domena se koristi u smislu skupa vrednosti iz kojeg semantički definisani objekti, kao što su tip entiteta i obeležje, uzimaju vrednosti.

S obzirom da, u opštem slučaju, obeležje uzima pojedine vrednosti iz svoga domena sa različitim verovatnoćama, obeležje se može smatrati semantički definisanom slučajnom veličinom. Tada se vrednosti, koje obeležje uzima, nazivaju njegovim konkretizacijama.

Konkretizacija obeležja predstavlja podatak o određenom entitetu i samo konkretizacija obeležja može predstavljati podatak.

Primer 1.16. Ako se posmatra obeležje $MATIČNI_BROJ_RADNIKA$ i jedna njegova konkretizacija 110451 , tada 110451 predstavlja podatak o nekom radniku jedino ako se unapred zna da 110451 predstavlja konkretizaciju obeležja $MATIČNI_BROJ_RADNIKA$. U suprotnom 110451 ne predstavlja podatak, jer mu je smisao neodređen. □

Formalno, neka je $E = \{e_i \mid i = 1, \dots, m\}$ klasa entiteta, a A jedno od obeležja te klase. Obeležje A predstavlja funkciju $A: E \rightarrow \text{dom}(A)$. Drugim rečima, obeležje A pridružuje svakom entitetu $e_i \in E$ jednu vrednost iz $\text{dom}(A)$, tako da $A(e_i)$ predstavlja podatak o e_i s obzirom na A .

Obeležje, koje se dalje ne može dekomponovati, ili koje se u posmatranom slučaju dalje ne dekomponuje na komponente, koje takođe predstavljaju obeležja, naziva se elementarnim obeležjem. Skup, niz, ili logički proizvod elementarnih obeležja predstavlja složeno obeležje. Tom nizu obeležja se može pridružiti neko ime.

Primer 1.17. Obeležja $NAZIV_PROIZVODA$, $BOJA_AUTOMOBILA$, $IME_STANOVNIKA$ predstavljaju elementarna obeležja različitih klasa entiteta. Složena obeležja predstavljaju, na primer, $ADRESA = \{MESTO, ULICA, BROJ\}$, $\{IME, PRZ, MESTO\}$, ili $DATUM_UPPLATE = \{DAN, MESEC, GODINA\}$. □

Složena obeležja se, često, označavaju slovima sa kraja abecede, na primer X ili Y ili Z , elementarna slovima sa početka abecede, na primer A ili B ili C . Saglasno rečenom, složeno obeležje je $X = \{A_1, A_2, \dots, A_k\}$, gde su A_i , $1 \leq i \leq k$, elementarna obeležja. Konkretizacija elementarnog obeležja predstavlja elementarni podatak, a konkretizacija složenog obeležja predstavlja složeni podatak. Ako je $X = \{A_1, \dots, A_k\}$, tada $\text{dom}(X) \subseteq \text{dom}(A_1) \times \dots \times \text{dom}(A_k)$. Obeležje čije se vrednosti dobijaju primenom nekog algoritma na vrednosti drugih obeležja naziva se **izvedenim obeležjem**, a njegove vrednosti **izvedenim podacima**.

Često se javlja potreba da se, pri evidentiranju podataka o nekom entitetu e , zabeleži da vrednost za neko obeležje A nedostaje. U tom cilju se u domen obeležja A uvodi fiktivna **nula vrednost**. Najčešće, nula vrednost ima jednu od sledeće dve semantike: "postojeća ali trenutno nepoznata vrednost", ili "obeležje A predstavlja neprimereno svojstvo za entitet e ". Nula vrednost se može zameniti bilo kojim elementom domena obeležja. U daljem tekstu, nula vrednost će se označavati sa ω .

→ TIP ENTITETA I POJAVA TIPOVIMA ENTITETA

Sa tačke gledišta zadataka informacionog sistema, nisu sva obeležja klase entiteta jednako važna. Od obeležja, bitnih za realizaciju zadataka informacionog sistema, gradi se **model** realne klase entiteta. Model klase entiteta naziva se **tipom entiteta**. Tip entiteta se označava sa $N(A_1, \dots, A_n)$, gde je N naziv tipa entiteta, A_i ($1 \leq i \leq n$) predstavljaju odabrana obeležja klase entiteta E . Kao i svaki model, tip entiteta predstavlja samo približnu sliku klase entiteta realnog sistema. Neki put se za reprezentaciju klase entiteta, umesto oznake za tip entiteta $N(A_1, \dots, A_n)$ koristi samo naziv N .

Pošto niz $N(A_1, \dots, A_n)$ predstavlja složeno obeležje, tip entiteta $N(A_1, \dots, A_n)$ predstavlja imenovano složeno obeležje.

Klasa entiteta poseduje konačno mnogo osobina zajedničkih svim realnim entitetima. Neka je $\{A_1, \dots, A_m\}$ skup osobina klase entiteta E . Tada je skup obeležja $\{A_1, \dots, A_n\}$ odabranih za izgradnju tipa entiteta kao modela klase E , pravi ili nepravi podskup skupa obeležja $\{A_1, \dots, A_m\}$.

Primer 1.18. Tip entiteta *STUDENT* (*BROJ_INDEKSA*, *IME*, *PREZIME*, *NAZIV_FAKULTETA*, *BROJ_ISPITA*) reprezentuje sve studente jednog univerziteta. □

Svaka klasa entiteta predstavlja skup sličnih entiteta. Svakom entitetu posmatrane klase odgovaraju konkretnе vrednosti obeležja klase entiteta. U okviru informacionog sistema, svaki entitet se predstavlja modelom koji sadrži konkretnе vrednosti onih obeležja, uz pomoć kojih je opisana klasa entiteta. **Uređenje podataka** u modelu entiteta diktirano je **uređenjem skupa obeležja** u tipu entiteta. Obeležjima u tipu entiteta odgovaraju konkretizacije tih obeležja (podaci) u modelu entiteta istim redom.

Model jednog entiteta naziva se **pojavom** odgovarajućeg tipa entiteta. Tip entiteta i pojava entiteta predstavljaju apstraktne opise konkretnih realnih entiteta. Tip entiteta predstavlja model entiteta na višem nivou apstrakcije nego pojava tipa entiteta.

Formalno, za dati tip entiteta $N(A_1, \dots, A_n)$ skup funkcija $\{A_i : E \rightarrow \text{dom}(A_i) \mid i = 1, \dots, n\}$ jednoznačno određuje funkciju $(A_1, \dots, A_n) : E \rightarrow \text{dom}(A_1) \times \dots \times \text{dom}(A_n)$, gde je E klasa entiteta, a $\text{dom}(A_1) \times \dots \times \text{dom}(A_n) = \{(a_1, \dots, a_n) \mid a_i \in \text{dom}(A_i), i = 1, \dots, n\}$. Funkcija (A_1, \dots, A_n) je definisana

na sa $(A_1, \dots, A_n)(e) = (A_1(e), \dots, A_n(e))$, gde je $e \in E$ entitet. Drugim rečima, (a_1, \dots, a_n) predstavlja pojavu tipa entiteta $N(A_1, \dots, A_n)$, ako je zadovoljena kvantifikatorska formula

$$\exists((a_1, \dots, a_n)) = (\exists e \in E)(i \in \{1, \dots, n\})(a_i = A_i(e)),$$

a skup P pojava tipa entiteta je

$$P = \{(a_1, \dots, a_n) \mid \exists((a_1, \dots, a_n))\}.$$

Uređenje podataka u n -torki (a_1, \dots, a_n) je bitno, jer nosi informaciju o tome da je $a_i \notin \text{dom}(A_i)$.

Primer 1.19. Za tip entiteta STUDENT iz primera 1.18, moguće pojave predstavljaju sledeće petorke:

- (159, Ivo, Ban, Fakultet tehničkih nauka, 15),
- (313, Eva, Tot, Ekonomski fakultet, 23),
- (505, Aco, Kon, Pravni fakultet, ω). □

Treba zapaziti da za studenta Aco Kon, obeležje BROJ_ISPITA ima nula vrednost. Moguća semantika te nula vrednosti je „postojeća ali trenutno nepoznata vrednost“. □

→ KLJUČ

Definicija pojma entiteta kao „jedinice posmatranja“ ukazuje na potrebu da se entiteti posmatrane klase mogu razlikovati. To zahteva da i modeli dva entiteta budu različiti. Neka su e_1 i e_2 entiteti klase E , a $N(A_1, \dots, A_n)$ tip entiteta, tada mora važiti $(A_1, \dots, A_n)(e_1) \neq (A_1, \dots, A_n)(e_2)$. Znači, mora postojati neprazan skup obeležja $X \subseteq \{A_1, \dots, A_n\}$ takav da je $X(e_1) \neq X(e_2)$.

Definicija 1.11. Neka je $P = \{p_i \mid i = 1, \dots, k\}$ skup svih pojava tipa entiteta $N(A_1, \dots, A_n)$, a $p[X]$ restrikcija pojave p na obeležje X . Obeležje X predstavlja *ključ* tipa entiteta $N(A_1, \dots, A_n)$ ako i samo ako važe sledeća tri uslova:

- 1⁰ $(\forall p_i, p_j \in P)(p_i \neq p_j \Rightarrow p_i[X] \neq p_j[X])$; *jedinstvenost*
- 2⁰ $(\forall X' \subset X)(\neg 1^0)$ i *minimalnost*
- 3⁰ $(\forall p \in P)(\forall A \notin X)(p[A] \neq \omega)$. □ *ne postoji nula vrednost*

Uslov 1⁰ se naziva uslovom jedinstvenosti i on ukazuje da sve pojave tipa entiteta N moraju imati različite vrednosti ključa X . Uslov 2⁰ se naziva uslovom minimalnosti i on ukazuje da ključ mora sadržati samo neophodna obeležja. Putem uslova 2⁰ se sprečava nepotrebna kompleksnost ključa. Konačno, uslov 3⁰ ukazuje da nijedna komponenta ključa ne sme imati nula vrednost. Uslov 3⁰ dopunjuje uslov 1⁰ u cilju obezbeđenja jedinstvenosti vrednosti ključa, jer nula vrednost predstavlja moguću zamenu za svaku vrednost iz domena obeležja. Ako X zadovoljava samo uslove 1⁰ i 3⁰, naziva se *superključem* tipa entiteta N .

Svaki tip entiteta poseduje bar jedan ključ. Neka je $X = \{A_1, \dots, A_n\}$ i važe uslovi 1⁰ i 3⁰. Tada je X superključ tipa entiteta N . Ako važi i uslov 2⁰, X predstavlja ključ tipa entiteta $N(A_1, \dots, A_n)$. Inače, mora postojati neprazan podskup skupa $\{A_1, \dots, A_n\}$ za koji važi 1⁰, 2⁰ i 3⁰. Ovo razmatranje, ujedno, sugerira i jedan od mogućih postupaka za određivanje ključa tipa entiteta.

Jedan tip entiteta može posedovati više ključeva. Nazivaju se *ekvivalentnim*. Jedan od ekvivalentnih ključeva se bira za *primarni*. U okviru notacije za tip entiteta sva obeležja jednog

ključa podvlače se jednom, kontinualnom linijom. Uobičajeno je da se, u definiciji tipa entiteta, sva obeležja - komponente jednog ključa podvuku kontinualnom linijom.

Primer 1.20. Tip entiteta *STUDENT(BROJ_INDEKSA, IME, PREZIME, ADRESA, MATIČNI_BROJ_STANOVNIKA)* poseduje dva ključa. To su *BROJ_INDEKSA* i *MATIČNI_BROJ_STANOVNIKA*. Ako je reč o tipu entiteta, definisanom u projektu fakultetskog informacionog sistema, obeležje *BROJ_INDEKSA* bi, nejverovatnije, bilo izabранo za primarni kluč.

Posmatra se tip entiteta *ISPIT(BROJ_INDEKSA, IDBROJ_PREDMETA, OCENA)*. Neka u realnom sistemu važe sledeća ograničenja:

1. jedan student može imati ocene iz različitih predmeta,
2. veći broj studenata može imati ocene iz istog predmeta i
3. svaki student ima najviše jednu ocenu iz jednog predmeta.

Skup obeležja *{BROJ_INDEKSA, IDBROJ_PREDMETA, OCENA}* predstavlja superključ tipa entiteta *ISPIT*. Na osnovu uslova 2. se zaključuje da obeležje *BROJ_INDEKSA* predstavlja obaveznu komponentu ključa, jer može postojati veći broj pojava tipa entiteta sa istim *{IDBROJ_PREDMETA, OCENA}* vrednostima. Na osnovu uslova 1. se zaključuje da i obeležje *IDBROJ_PREDMETA* predstavlja obaveznu komponentu ključa, jer može postojati veći broj pojava tipa entiteta sa istim *{BROJ_INDEKSA, OCENA}* vrednostima. Na osnovu uslova 3. se zaključuje da se obeležje *OCENA* može izostaviti iz superključa, te se zaključuje da kluč predstavlja složeno obeležje *{BROJ_INDEKSA, IDBROJ_PREDMETA}*.

Prepostavimo da obeležje *IDBROJ_PREDMETA* sme imati nula vrednosti. Tada skup pojava tipa entiteta *ISPIT* može sadržati sledeće dve pojave (159, Matematika, 9) i (159, α, 9). Pošto nula vrednost predstavlja zamenu za svaku vrednost iz domena obeležja, ove dve pojave se, strogo govoreći, ne mogu razlikovati.

Tip entiteta *POVERAVANJE(IDBROJ_NASTAVNIKA, IDBROJ_PREDMETA)* opisuje odnos između nastavnika i predmeta u realnom sistemu, gde se nastavniku poverava izvođenje predavanja iz predmeta. Ako se za kluč tipa entiteta *POVERAVANJE* odredi obeležje *IDBROJ_NASTAVNIKA*, tada tip entiteta opisuje odnos, prema kojem se svakom nastavniku poverava samo jedan predmet. Ako se za kluč odredi *IDBROJ_PREDMETA*, tip entiteta opisuje realnu situaciju u kojoj nastavu iz svakog predmeta izvodi samo jedan nastavnik.

Ako se prepostavi da se istom nastavniku može poveriti izvođenje nastave iz više predmeta, a isti predmet može poveriti većem broju nastavnika, tada sva obeležja tipa entiteta *POVERAVANJE (IDBROJ_NASTAVNIKA, IDBROJ_PREDMETA)* predstavljaju kluč. Može se zaključiti da semantika modela klase entiteta značajno zavisi od odabranog kluča. □

TIP SLOGA I POJAVA TIPOA SLOGA

Bez pretenzije na strogost definicije, smatraće se da pojmovi tip sloga i pojava tipa sloga predstavljaju sinonime, redom, za tip entiteta i pojavu tipa entiteta. Saglasno tome i skup P pojava tipa entiteta predstavlja skup pojava tipa sloga definisanog nad istim skupom obeležja kao i tip entiteta.

DATOTEKA

Generalno govoreći, datoteke se mogu podeliti na:

- programske datoteke, koje sadrže programe bilo u izvornom ili mašinskom kodu i
- datoteke, koje sadrže podatke.

Datoteke koje sadrže podatke se mogu dalje deliti na:

- datoteke sa podacima u slobodnom formatu (polustrukturirane i nestrukturirane datoteke) i
- datoteke sa veoma strogim formatom (strogo definisanom strukturu).

Samo datoteke sa strogim formatom podataka predstavljaju predmet izučavanja u ovoj knjizi. Saglasno tom opredeljenju se, u daljem tekstu, pojam datoteke i definiše.

Datoteka F predstavlja strukturu nad skupom P pojava tipa sloga $N(A_1, \dots, A_n)$. Relacija R , koja uređuje skup P , najčešće se definiše s obzirom na vrednosti primarnog ključa tipa sloga N . Saglasno rečenom, $F = (P, R)$. Datoteka poseduje ime i nalazi se memorisana na medijumu nekog memorijskog uređaja. Najčešće, datoteci se dodeljuje naziv odgovarajućeg tipa sloga.

Tip sloga i datoteka predstavljaju modele klase entiteta. Pri tome, tip sloga predstavlja model na višem nivou apstrakcije. Istovremeno, tip sloga predstavlja i apstraktni model odgovarajuće datoteke.

Primer 1.21. Sledeće pojave tipa sloga

ISPIT(BROJ_INDEKSA, PREDMET, PREZIME, OCENA):

((013, Matematika, Marić, 9),
(013, Mehanika, Marić, 7),
(117, Matematika, Marić, 7),
(159, Fizika, Ban, 6),
(159, Matematika, Ban, 7))

reprezentuju jednu od mogućih datoteka nad tim tipom sloga. □

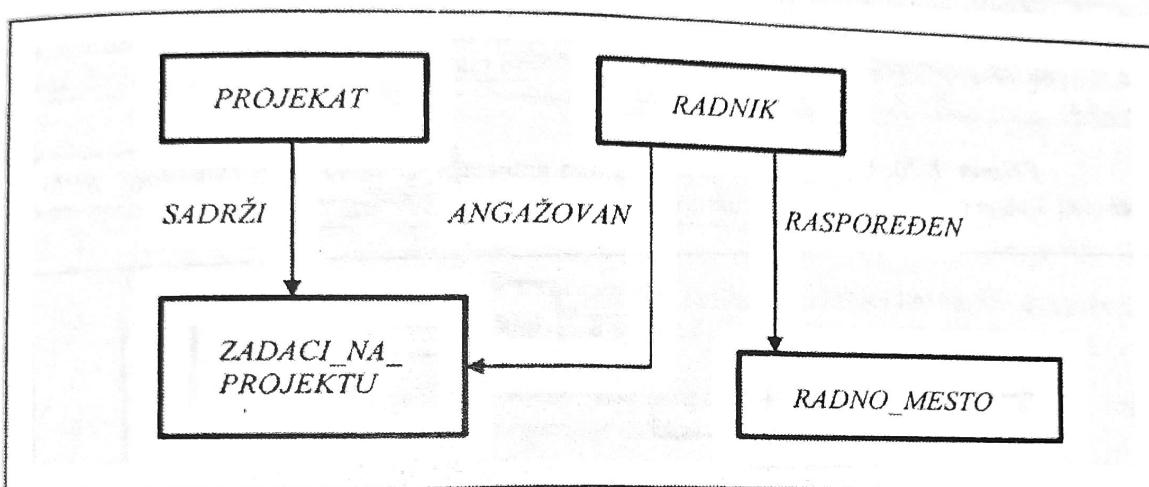
1.5. LOGIČKE STRUKTURE NAD SKUPOM OBELEŽJA

Projektovanje automatizovanog informacionog sistema vrši se na osnovu prethodno definisanih zahteva, koje taj informacioni sistem treba da zadovolji. Automatizovani informacioni sistem se projektuje sa ciljem obezbeđenja uslova za efikasnije upravljanje razvojem i operativnim funkcionisanjem nekog realnog sistema. Taj realni sistem može predstavljati neka radna organizacija, društveno-politička zajednica, neka druga asocijacija građana ili neki njihov deo. Kada je reč o projektovanju organizacije podataka tog informacionog sistema, postoji više karakterističnih grupa zahteva. Na ovom mestu će biti ukazano samo na dve. To su:

- grupa zahteva koja ukazuje na to, o kojim klasama entiteta realnog sistema treba da se iskazuju podaci u izlazima informacionog sistema (grupa zahteva Z1) i
- grupa zahteva koja ukazuje na to, koji podaci o entitetima pojedinih klasa treba da se iskazuju (grupa zahteva Z2).

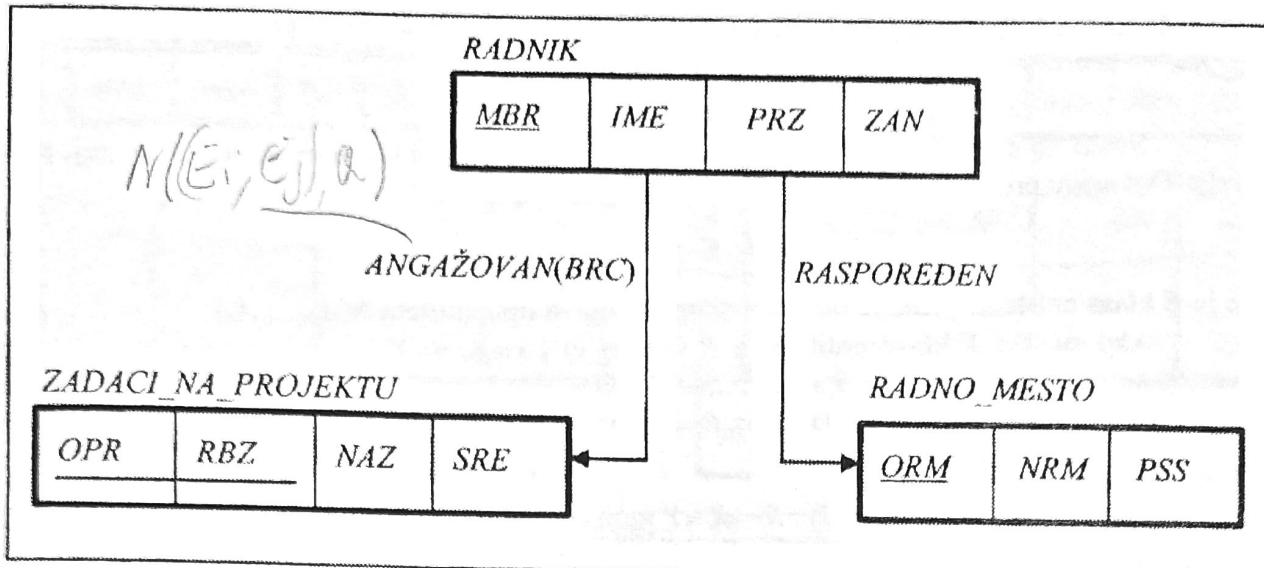
Na osnovu zahteva iz grupe Z1 gradi se, u postupku projektovanja organizacije podataka, graf $G = (S, R)$, gde je $S = \{E_i \mid i = 1, \dots, n\}$ skup klasa entiteta sa nazivima (oznakama) E_1, \dots, E_n , a $R = \{(E_i, E_j) \mid E_i, E_j \in S \text{ i } E_i \text{ je u vezi sa } E_j\}$ relacija, koja ukazuje da između entiteta klase E_i i klase E_j u realnom sistemu postoji neka veza. Pri crtanju grafa G , nazivi klasa entiteta predstavljaju identifikatore čvorova grafa, a često se i ivicama (granama) grafa pridružuju identifikatori koji opisuju vezu između entiteta povezanih klasa.

Primer 1.22. Na slici 1.15 je nacrtan graf nad skupom klasa entiteta jedne hipotetične projektne organizacije. Graf sa slike 1.15 se može opisati sledećim rečenicama. Radnici su raspoređeni na radna mesta i angažovani na zadacima u okviru nekog projekta. Projekti sadrže zadatke. □



Slika 1.15.

Neka je U skup svih obeležja realnog sistema. Na osnovu zahteva iz grupe Z2, identifikuju se ona obeležja skupa U koja su bitna za realizaciju zadataka informacionog sistema. Ova identifikacija predstavlja jedan od koraka metodologije projektovanja organizacije podataka. Jedan drugi korak ove metodologije, recimo korak L , pridružuje svakom čvoru i svakoj ivici grafa G jedan podskup skupa obeležja U . Drugim rečima, L predstavlja funkciju koja preslikava S i R u partitivni skup skupa U , tj $L : S \rightarrow \mathcal{P}(U)$ i $L : R \rightarrow \mathcal{P}(U)$. Pri tome, L pridružuje čvorovima iz S neprazne podskupove skupa U , dok nekim ivicama iz R funkcija L može pridružiti i prazan skup. Očigledno, funkcija L pridružuje čvorovima i ivicama grafa G semantiku.



Slika 1.16.

Primer 1.23. Na slici 1.16 prikazan je jedan podgraf grafa sa slike 1.15 čijim čvorovima i ivicama je pridružena semantika. Čvoru *RADNIK* pridružena su obeležja: matični broj radnika (*MBR*), ime (*IME*), prezime (*PRZ*) i zanimanje (*ZAN*). Čvoru *ZADACI NA PROJEKTU* pridružena su obeležja: oznaka projekta (*OPR*), redni broj zadatka na projektu (*RBZ*), naziv zadatka (*NAZ*), sredstva na zadatku (*SRE*). Čvoru *RADNO MESTO* pridružena su obeležja: oznaka radnog mesta (*ORM*), naziv radnog mesta (*NRM*) i potrebna stručna spremna (*PSS*). Ivici *ANGAŽOVAN* pridruženo je obeležje broj časova rada radnika na zadatku (*BRC*), a ivici *RASPOREĐEN* nije pridruženo nijedno obeležje. □

U skup obeležja jednog čvora, ili ivice, često se uvodi poredak. Taj poredak ne nosi nikakvu semantiku. Uvođenjem poretku, svaki čvor sa pridruženom semantikom postaje tip entiteta. Bitan korak metodologije projektovanja organizacije podataka predstavlja i određivanje ključeva tipova entiteta, o čemu je već bilo reći.

Svaka ivica grafa G sa pridruženom semantikom predstavlja imenovani objekat čije komponente su povezane klase entiteta i skup pridruženih obeležja Q , u oznaci $N((E_i, E_j), Q)$. Ovaj objekat se naziva tipom poveznika. Kao i u slučaju tipa entiteta, uobičajeno je da se u skup Q uvede poredak.

Primer 1.24. Na slici 1.16 su prikazana dva tipa poveznika *ANGAŽOVAN* ((*RADNIK*, *ZADACI NA PROJEKTU*), *BRC*) i *RASPOREĐEN*(*RADNIK*, *RADNO MESTO*). □

Pošto tipovi entiteta i tipovi poveznika predstavljaju imenovana složena obeležja, graf G snabdeven semantikom na nivou apstrakcije obeležja se naziva i logičkom strukturom nad skupom obeležja. Takođe, ovaj graf predstavlja statički model realnog sistema i model buduće baze podataka. Svaki tip entiteta predstavlja model jedne klase entiteta, a tip poveznika model odnosa između entiteta dve klase. Tip entiteta i tip poveznika predstavljaju i logičke strukture nad skupom obeležja i modele odgovarajućih datoteka. Graf G kod kojeg je kardinalitet skupa S veći od 1 predstavlja statički model realnog sistema i baze podataka informacionog sistema tog realnog sistema.



1.6 LOGIČKE STRUKTURE NAD SKUPOM PODATAKA

Svaki tip entiteta kao funkcija preslikava odgovarajuću klasu entiteta u skup koji predstavlja Dekartov proizvod domena obeležja tipa entiteta, tj.

$$(A_1, \dots, A_n) : E \xrightarrow{\circ} \underline{dom(A_1) \times \dots \times dom(A_n)},$$

gde je E klasa entiteta. Time je određen skup P pojava tipa entiteta $N(A_1, \dots, A_n)$.

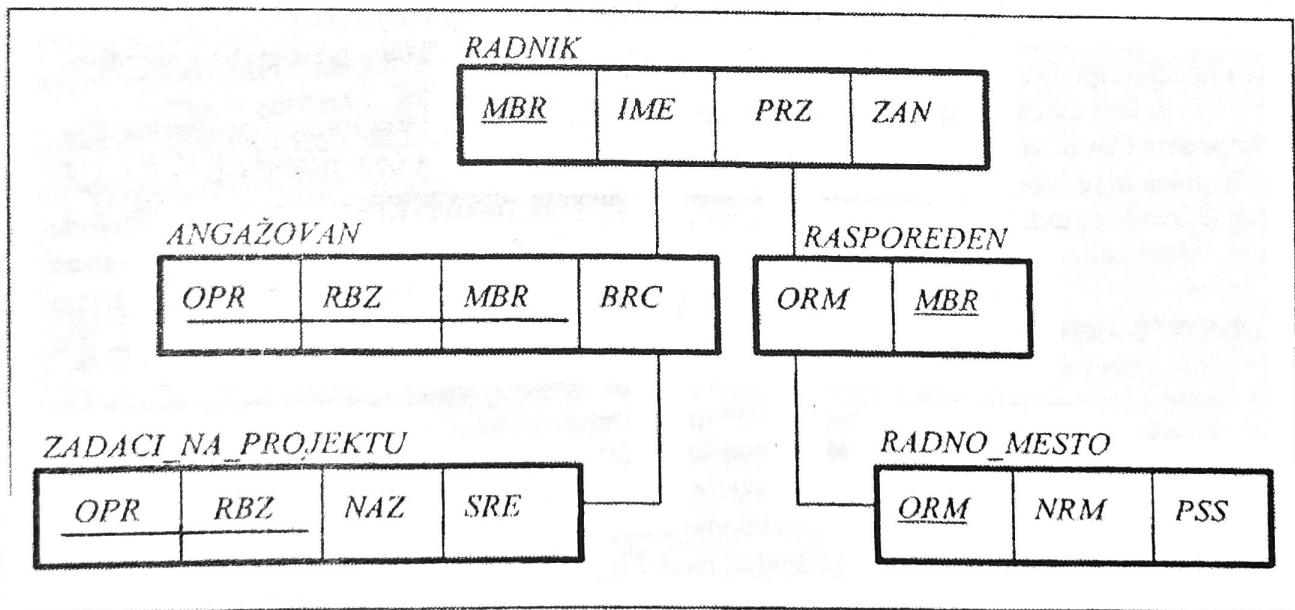
Ako su X i Y klase entiteta, a $R = \{(x, y) \mid x \in X, y \in Y \text{ i } x \text{ je u vezi sa } y\}$, tada tip poveznika $N(X, Y, Q)$, gde je $Q = (A_1, \dots, A_k)$, eventualno, prazan niz obeležja, pridružuje semantiku svakom uređenom paru (x, y) iz R . **Tip poveznika** $N(X, Y, Q)$ predstavlja funkciju koja preslikava R u skup torki reda $k + 2$, tj.

$$(X, Y, Q) : R \xrightarrow{\circ} \underline{X \times Y \times dom(A_1) \times \dots \times dom(A_k)}.$$

Torka (x, y, a_1, \dots, a_k) , gde $a_i \in dom(A_i)$, predstavlja **pojavu tipa poveznika** $N(X, Y, Q)$. Pri tome, x i y su entiteti.

Neka je K primarni ključ tipa entiteta $N(A_1, \dots, A_n)$, pridruženog klasu entiteta E . Preslikavanje $K : E \xrightarrow{\circ} P$, gde je P skup pojava tipa entiteta, je bijektivno. Saglasno tome, kao reprezent svakog entiteta e iz E može se koristiti vrednost ključa iz odgovarajuće pojave tipa entiteta. Dalje, pojava tipa poveznika (x, y, a_1, \dots, a_k) može se transformisati u torku $(k_x, k_y, a_1, \dots, a_k)$, gde su k_x i k_y vrednosti ključeva tipova entiteta pridruženih, redom, klasama entiteta X i Y .

Ako se pretpostavi da u jednoj logičkoj strukturi nad skupom obeležja ne postoje dva tipa entiteta sa istim ključem, tada se i tip poveznika $N(X, Y, A_1, \dots, A_k)$ može transformisati u $N(K_x, K_y, A_1, \dots, A_k)$, gde su K_x i K_y primarni ključevi tipova entiteta pridruženih redom klasama X i Y . Ključ tipa poveznika $N(K_x, K_y, A_1, \dots, A_k)$ je $K \subseteq K_x \cup K_y$.



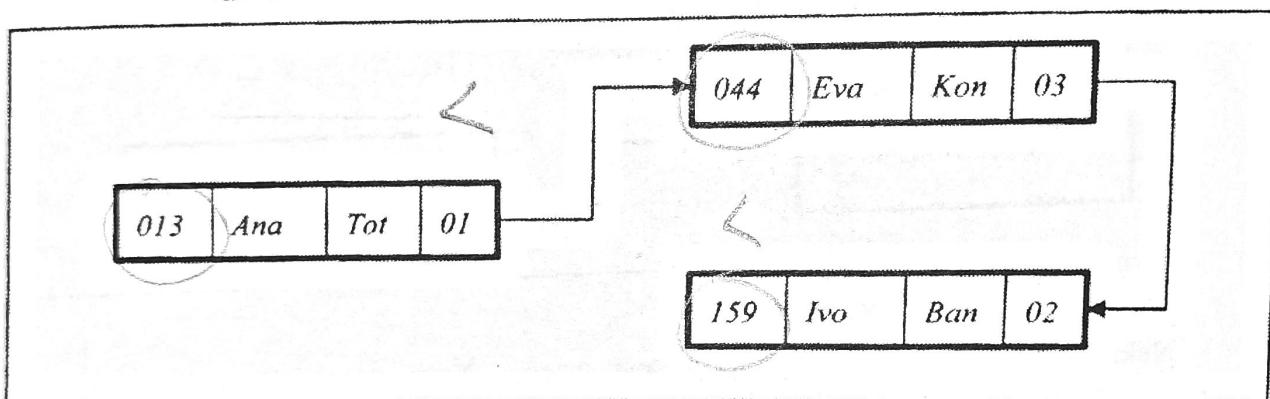
Slika 1.17.

Primer 1.25. Na slici 1.17 je nacrtana ista logička struktura nad skupom obeležja kao i na slici 1.16, samo je za predstavljanje tipova poveznika upotrebljena geometrijska reprezentacija koja se koristi za predstavljanje tipa entiteta. Pod pretpostavkom da jedan radnik može biti raspoređen na samo jedno radno mesto, ključ tipa poveznika *RASPOREDEN* je samo MBR. Neorientisanim potezima na slici 1.17 nije pridružena semantika. □

Definicija 1.12. Kada se u skup P pojava tipa entiteta $N(A_1, \dots, A_n)$ ili tipa poveznika $N(K_x, K_y, A_1, \dots, A_k)$ uvede relacija strogog poretku sa semantikom "ima veću (manju) vrednost ključa od", dobija se *logička struktura podataka datoteke*. □

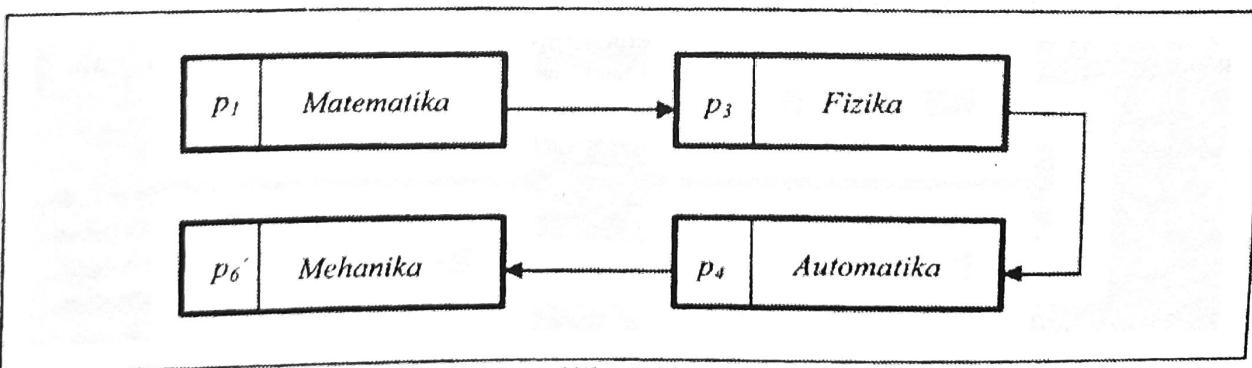
Znači, logičku strukturu podataka datoteke predstavlja graf $G = (S, R)$ sa pridruženom semantikom, gde je S skup pojava jednog tipa entiteta, ili skup pojava jednog tipa poveznika, a R relacija strogog porekla, koja skup S ureduje saglasno rastućim (ili opadajućim) vrednostima ključa.

Primer 1.26. Neka je S skup pojava tipa entiteta **STUDENT(BRI, IME, PRZ, BPI)**. Na slici 1.18 je prikazana logička struktura podataka datoteke STUDENT. □



Slika 1.18.

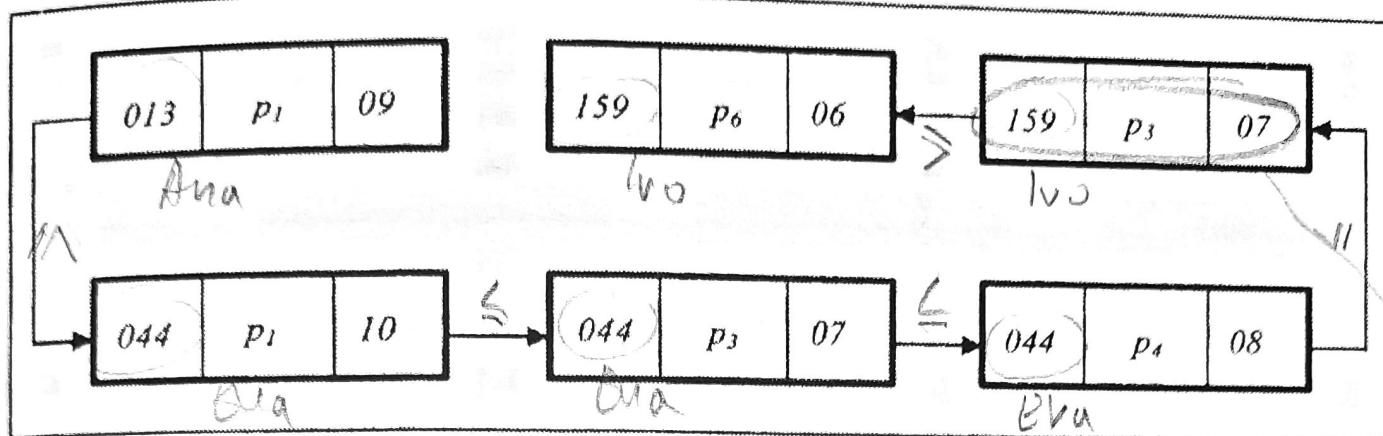
Primer 1.27. Neka je S skup pojava tipa entiteta **PREDMET(OZP, NAZ)**. Na slici 1.19 je prikazana logička struktura podataka datoteke PREDMET. □



Slika 1.19.

$$p_1 < p_3 < p_4 < p_6$$

Primer 1.28. Neka je X skup pojava tipa entiteta **STUDENT** iz primera 1.26, Y skup pojava tipa entiteta **PREDMET** iz primera 1.27, a **ISPIT(BRI, OZP, OCE)** tip poveznika. Na slici 1.20 je prikazana logička struktura podataka datoteke **ISPIT**. □

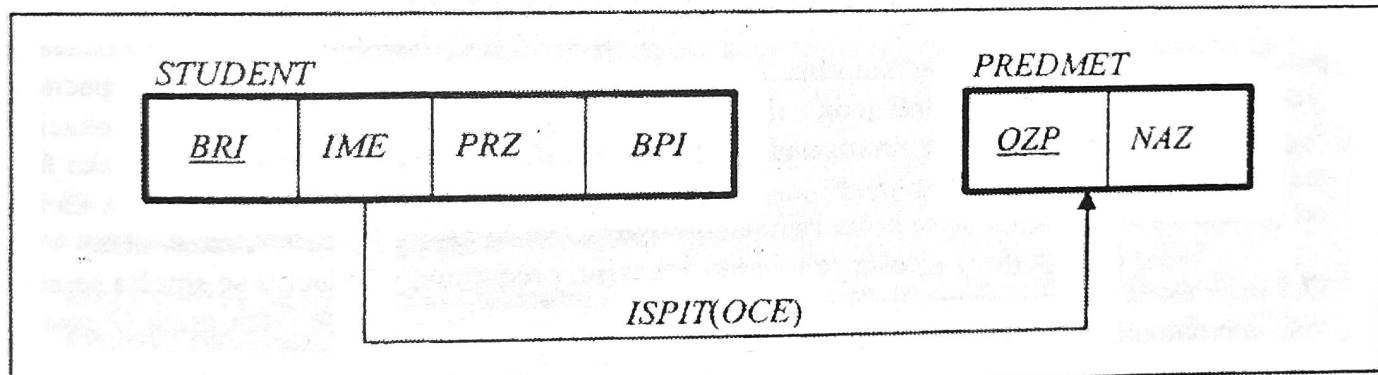


Slika 1.20.

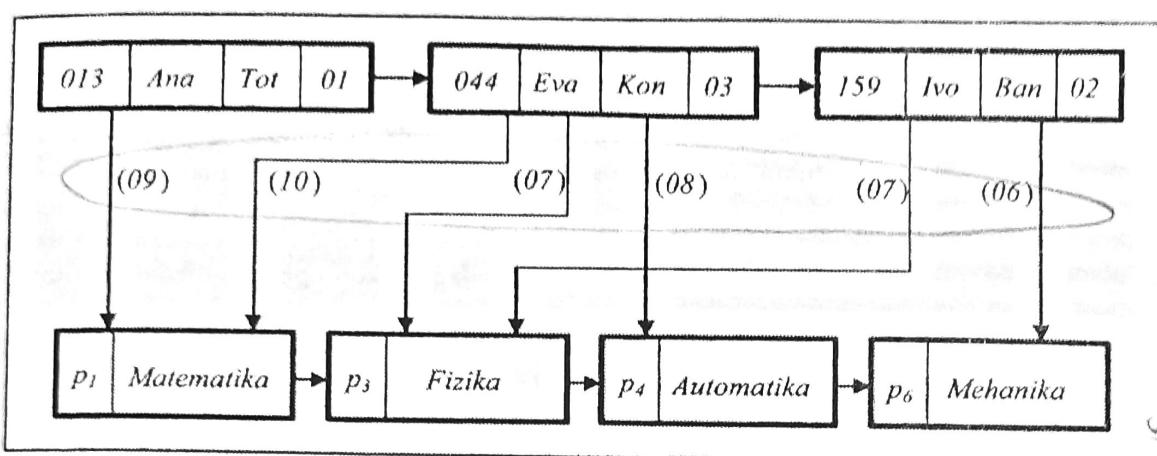
Neka je *LSO* logička struktura nad skupom obeležja, dobijena u koraku L projektovanja organizacije podataka.

Definicija 1.13. Logičku strukturu nad skupom podataka baze podataka predstavlja graf $G = (S, R)$ čijim čvorovima i ivicama je putem *LSO* pridružena semantika. □

Primer 1.29. Logička struktura nad skupom obeležja sa slike 1.21 definiše logičku strukturu podataka baze podataka, prikazanu na slici 1.22. □



Slika 1.21.



Slika 1.22.

Opisane strukture nad skupom podataka nazivaju se logičkim, jer predstavljaju sliku dela realnog sveta koja, po pravilu, egzistira samo u mislima projektanta, programera i korisnika datoteke ili baze podataka. Logičke strukture nad skupom podataka predstavljaju modele realnih sistema i njihovih baza podataka na nivou apstrakcije podataka.

Sama logička struktura nad skupom podataka je vremenski promenljiva struktura, ona prati promene u realnom sistemu. Za razliku od nje, logička struktura nad skupom obeležja se može smatrati uslovno staticnom. Tek značajne strukturalne promene u realnom sistemu dovođe do promena i u logičkoj strukturi nad skupom obeležja.

Za prikazivanje logičkih struktura nad skupom podataka, pored usmerenih grafova, često se koriste i tabele sa homogenim kolonama. Za skup pojava svakog tipa entiteta ili tipa poveznika koristi se posebna tabela. Iznad tabele se upisuje naziv tipa entiteta ili poveznika, u zaglavlje tabele se upisuju obeležja tipa entiteta ili poveznika, a u vrste tabele unose se pojave.

Primer 1.30. Na slici 1.23 je data tabelarna predstava logičke strukture nad skupom podataka iz primera 1.29. □

BRI	IME	PRZ	BPI
159	Ivo	Ban	02
013	Ana	Tot	01
044	Eva	Kon	03

OZP	NAZ
p1	Matematika
p3	Fizika
p6	Mehanika
p4	Automatika

BRI	OZP	OCE
159	p3	07
013	p1	09
044	p3	07
044	p1	10
159	p6	06
044	p4	08

Slika 1.23.

BASE
FOR AKA
DBA

DBA
BASE

1.7 FIZIČKE STRUKTURE PODATAKA

Kada se semantika, pridružena čvorovima i ivicama logičke strukture nad skupom podataka smesti na medijum memoriskog uređaja, dobija se fizička struktura podataka. Pri tome se, za predstavljanje podataka, umesto prirodne koristi binarna azbuka. Pored podataka koje sadrži logička struktura, fizička struktura podataka se proširuje i podacima svojstvenim postupcima memorisanja na konkretnom uređaju, podacima koji opisuju određene osobine pojava tipova entiteta (na primer: dužina pojave izražena brojem znakova prirodne azbuke, pri-padnost tipu entiteta i slično). Takođe, postoje različiti postupci za predstavljanje ivica logičke strukture nad skupom podataka na memoriskom medijumu.

