ALGORITMI I STRUKTURA PODATAKA

Studijski programi:

Softversko inženjerstvo

Računarska tehnika

Matematika-informatika

Nastava

- Predavanja (2 časa)

  Ulfeta Marovac

  e-mail: umarovac@np.ac.rs
- Vežbe (2 časa)

  Aldina Avdić

  e-mail: apljaskovic@np.ac.rs

Poeni

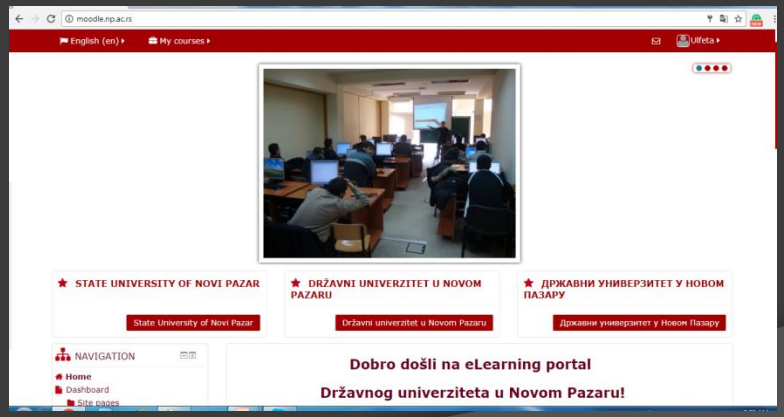
Domaći zadatak	Ocena od 6 do 10
Kolokvijum (praktični deo)	40
Pismeni ispit	60
Usmeni ispit	100

Poeni sa praktičnog dela ispita i domaćih zadataka se čuvaju do kraja školske godine.

Studenti koji ne polažu ispit preko kolokvijuma na pismenom ispitu mogu osvojiti 100 poena.

Literatura

- Milo Tomašević, Algoritmi i strukture podataka
- Dejan Živković,Uvod u algoritme i strukture podataka
- Sva dodatna literatura biće postavljena na portalu za elektronsko učenje Državnog univerziteta u Novom Pazaru.
- moodle.np.ac.rs



Termini predavanja i konsultacija

- Predavanja : ponedeljak, 15h, amfitetar A1
- Konsultacija: ponedeljak i utorak, 10-12h, kabinet 303

Ocene

BROJ POENA	OCENA
51-60	6
61-70	7
71-80	8
81-90	9
91-100	10

Cilj predmeta

- Cilj predmeta je da se student uvede u osnovne strukture podataka i da ovlada različitim tehnikama implementacije struktura, optimizacije algoritama i primene struktura.

Sadržaj

- Osnove o algoritmima i strukturama podataka
- Analiza algoritma
- Pokazivači
- Nizovi i matrice
- Ulančane liste
- Stek
- Redovi
- Heširanje
- Stabla
- Grafovi
- Pretraživanje
- Sortiranje

Algoritmi i strukture podataka

- Algoritmi i strukture čine osnovne dva osnovna gradivna bloka za implementaciju programskih sistema.
- Strukture podataka se koriste za opis načina organizacije podataka.
- Algoritmi služe za opis načina obrade podataka.

Podatak

- Podatak je osnovni element svakog programskog sistema i manipulisanje podacima je osnovna njegova svrha.
- Tip podataka- skup vrednosti i operacija čije izvršavanje je dozoljeno nad skupom datih vrednosti.
- Svaki programski jezik definiše neke osnovne primitivne tipove podataka koji se sastoje od atomičnih nedeljivih vrednosti.
- Apstraktni tipovi podataka su generalizacije primitivnih tipova podataka.
- Apstrakcija podataka predstavlja logički opis kolekcije podataka i operacija koje sa nad tim podacima mogu izvršiti.

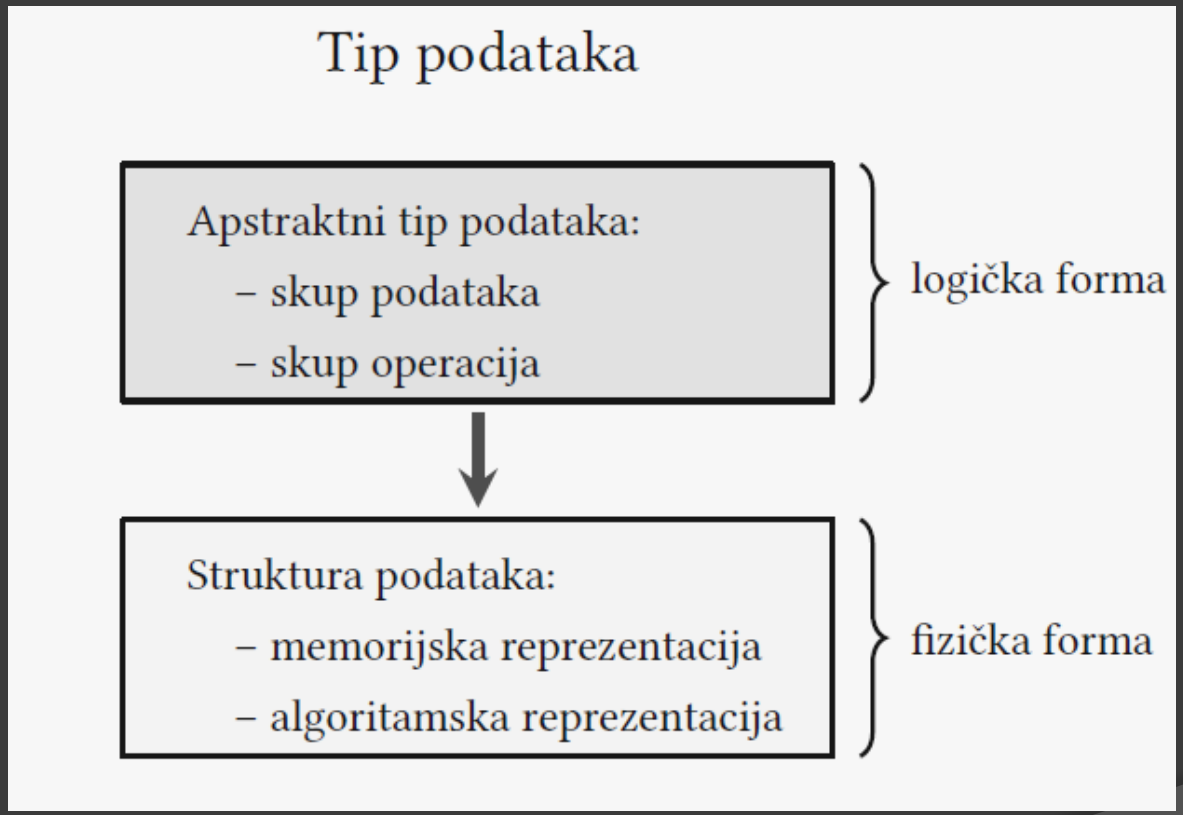
Definisanje logičkog modela podatka i njegova realizacija na računaru

1. Način modeliranja entiteta problema kao apstraktnog matematičkog objekta
2. Definisanje skupa dozvoljenih operacija nad tim objektima
3. Način predstavljanja u memoriji računara
4. Kolekcija algoritama na kojima se zasniva programska realizacija dozvoljenih operacija nad ovim objektima

Struktura podataka

- Pojam apstraktnog tipa podataka obuhvata elemente 1. i 2. koji određuju domen matematički definisanih objekata i skup operacija koje se mogu primeniti nad njima.
- Strukture podataka sadrže elemente 3. i 4. kojima se određuje kako su ovi apstraktni matematički objekti implementirani u računaru.

Logička i fizička forma podatka

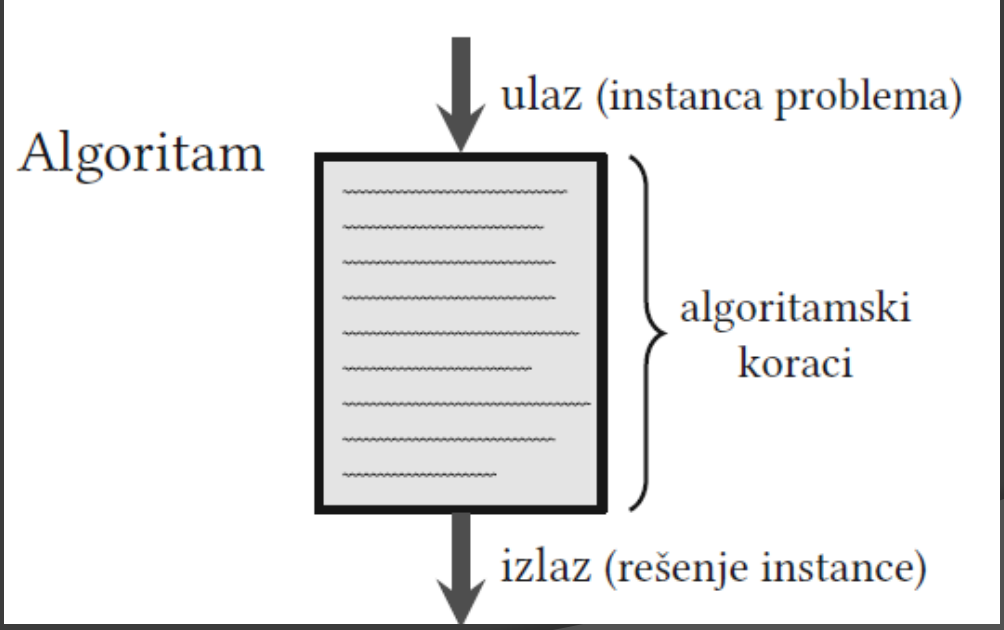


Algoritmi

- Neformalno govoreći, algoritam je precizan opis postupka za rešavanje nekog problema u konačnom broju koraka.
- U računarstvu algoritam je precizno definisana računarska procedura koja pretpostavlja neke podatke kao ulaz i proizvodi neke podatke kao izlaz.

Algoritam

- Algoritam se sastoji od jasne i nedvosmislene specifikacije niza koraka koji se mogu mehanički izvršiti na računaru.



Odlike algoritma

- Algoritam mora biti postupak koji se satsoji od konačno mnogo koraka koji se mogu izvršiti na računaru.
- Algoritam mora biti postupak po kojem je nedvosmisleno određen svaki sledeći korak za izvršavanje.
- Algoritam mora biti ispravan. Za svaku instancu problema na ulazu algoritma mora se dobiti odgovarajući izlazni rezultat.

Algoritam

- Specifikacija algoritma može biti izražena:
  - Opisivanjem rečima prirodnog jezika
  - U grafičkoj formi dijagramom toka podataka
  - U nekom programskom jeziku.

Blok dijagram algoritma

- Najstarija i najrasprostranjenija, grafička notacija za predstavljanje algoritama.
- Uvedena 60-tih godina kao osnovno sredstvo za opis algorima koji će biti implementirani na programskom jeziku Fortran.
- Grafički simboli blok dijagrama i tehnika njihove upotrebe regulisana je međunarodnim standardom ISO R 1028

Osnovni grafički simboli blok dijagrama

- Terminalni simbol označava početak, odnosno kraj algoritma. Ispunjen je rečima POČETAK i KRAJ (START i STOP).

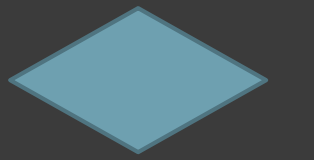
- Simbol obrade označava operaciju koja će se izvršiti u datom trenutku izvršavanja algoritma. Operacija se navodi unutar simbola.

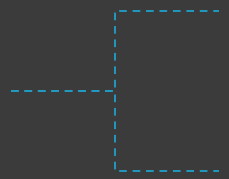
- Simbol ulaza/izlaza označava ulaz izlaz vrednosti u/iz promenljivih navedenih unutar simbola. Obavezno naznačiti da li se radi o ulazu (ULAZ ili INPUT) ili izlazu (IZLAZ ili OUTPUT).

- Simbol ulaza (INPUT)

- Simbol izlaza (OUTPUT)

Osnovni grafički simboli blok dijagrama





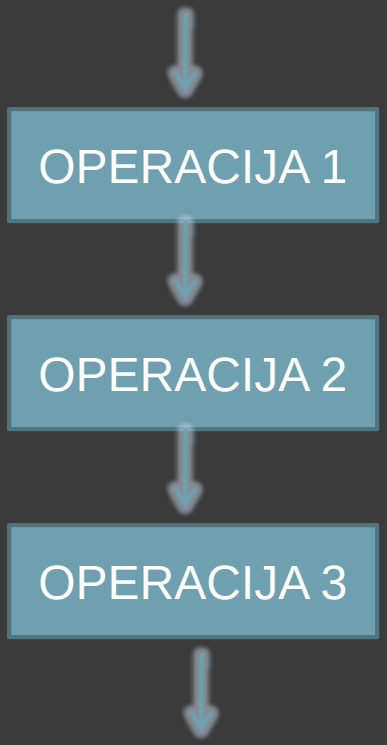
- Simbol selekcije (testa) označava grananje toka algoritama. Uslov se navodi unutar simbola a rezultati testiranja uslova se navode uz odgovarajuće tokove podataka
- Simbol poziva podprograma čiji su identifikatori i lista stvarnih parametara navedeni unutar simbola
- Simbol toka označava tok izvršavanja algoritma
- Simbol komentara, dodatnog objašnjenja objekta za koji je vezan
- Simbol konekcije omogućava konekciju više ulaznih tokova u jedan izlazni tok.

Elementi struktuiranog algoritma

- Rešenje bilo kog problema koji je po svojoj prirodi rešiv pomoću računara , može se izraziti kao superpozicija sledećih struktura:
  - sekvence
  - selekcije i
  - iteracije.

SEKVENCA

- Sekvenca je uređen niz instrukcija gde se i-ta instrukcija izvršava posle i-1 instrukcije
- Sekvenca se formira nizom simbola povezanih tokovima



SELEKCIJA

- Selekcija omogućava izbor jedne putanje kojom će se nastaviti izvršavanje instrukcija. Izbor putanje se vrši na osnovu uslova koji je definican kao logički izraz.
- Razlikujemo sledeće tipove selekcije:
  - **IF** *uslov* **THEN** *operacija*
  - **IF** *uslov* **THEN** *operacija1* **ELSE** *operacija2*
  - **CASE** *uslov*

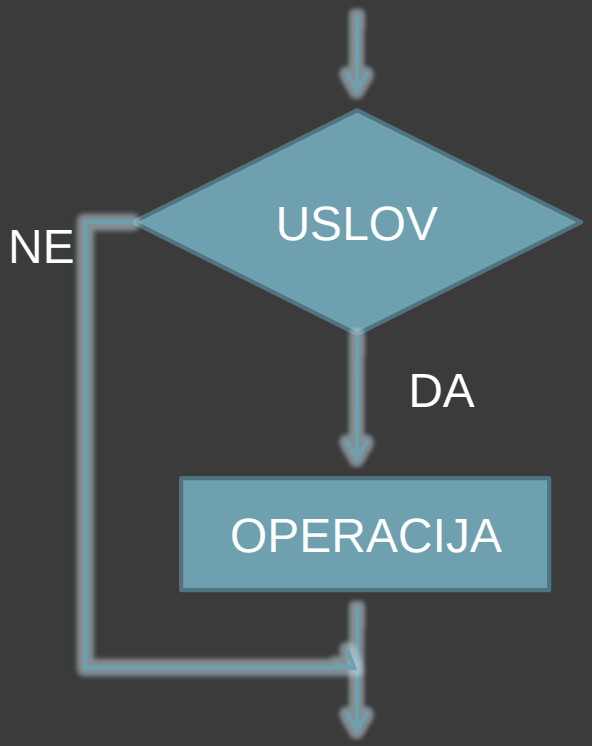
    OF V_1***: operacija1***

    OF V_2***: operacija2***

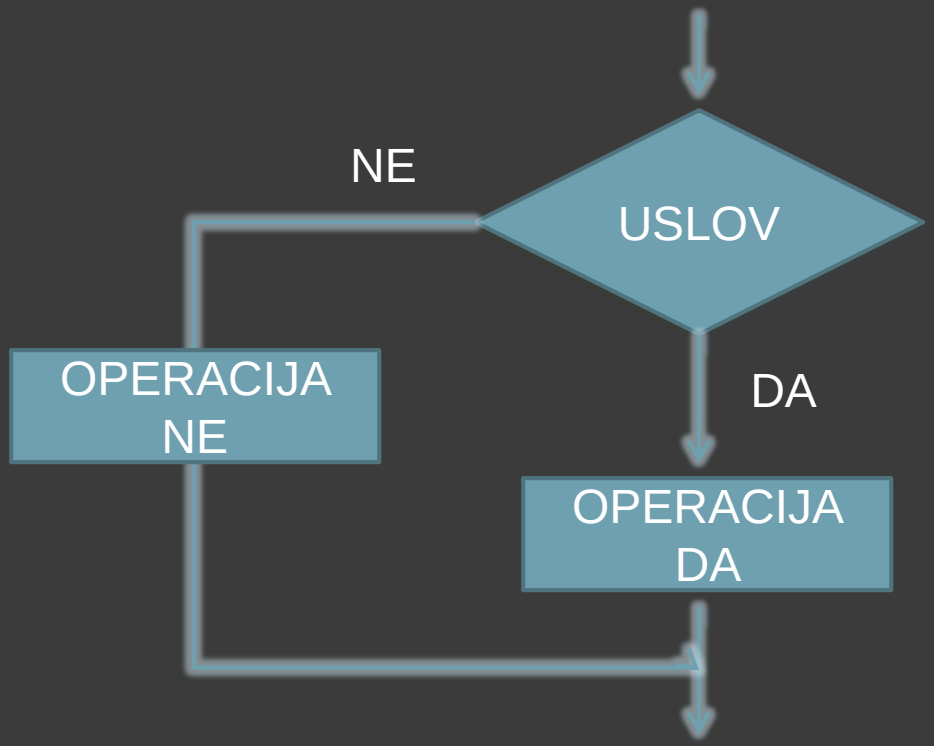
    ….

    ELSE operacijan

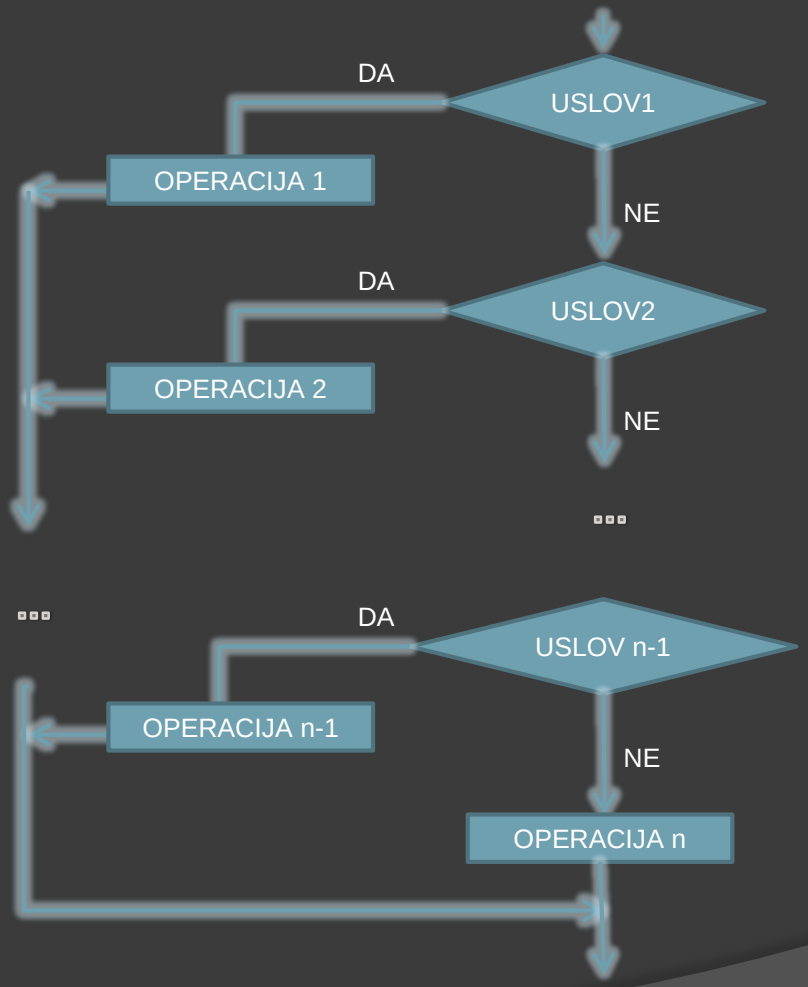
IF THEN SELEKCIJA



IF THEN ELSE SELEKCIJA



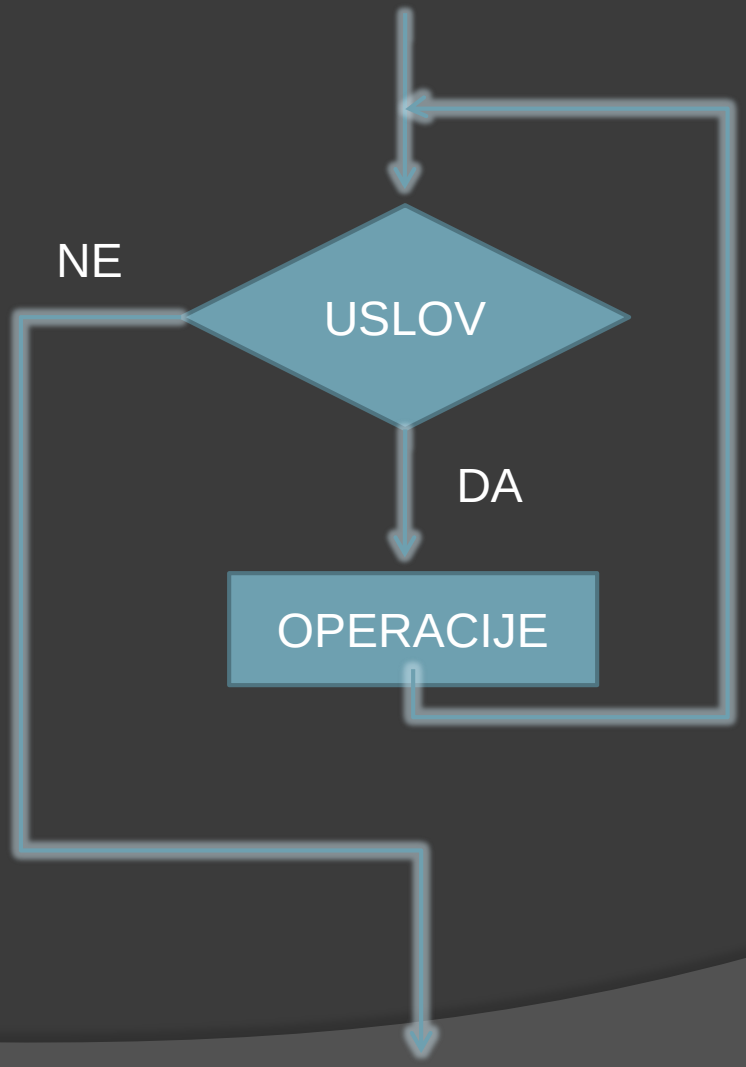
CASE OF SELEKCIJA



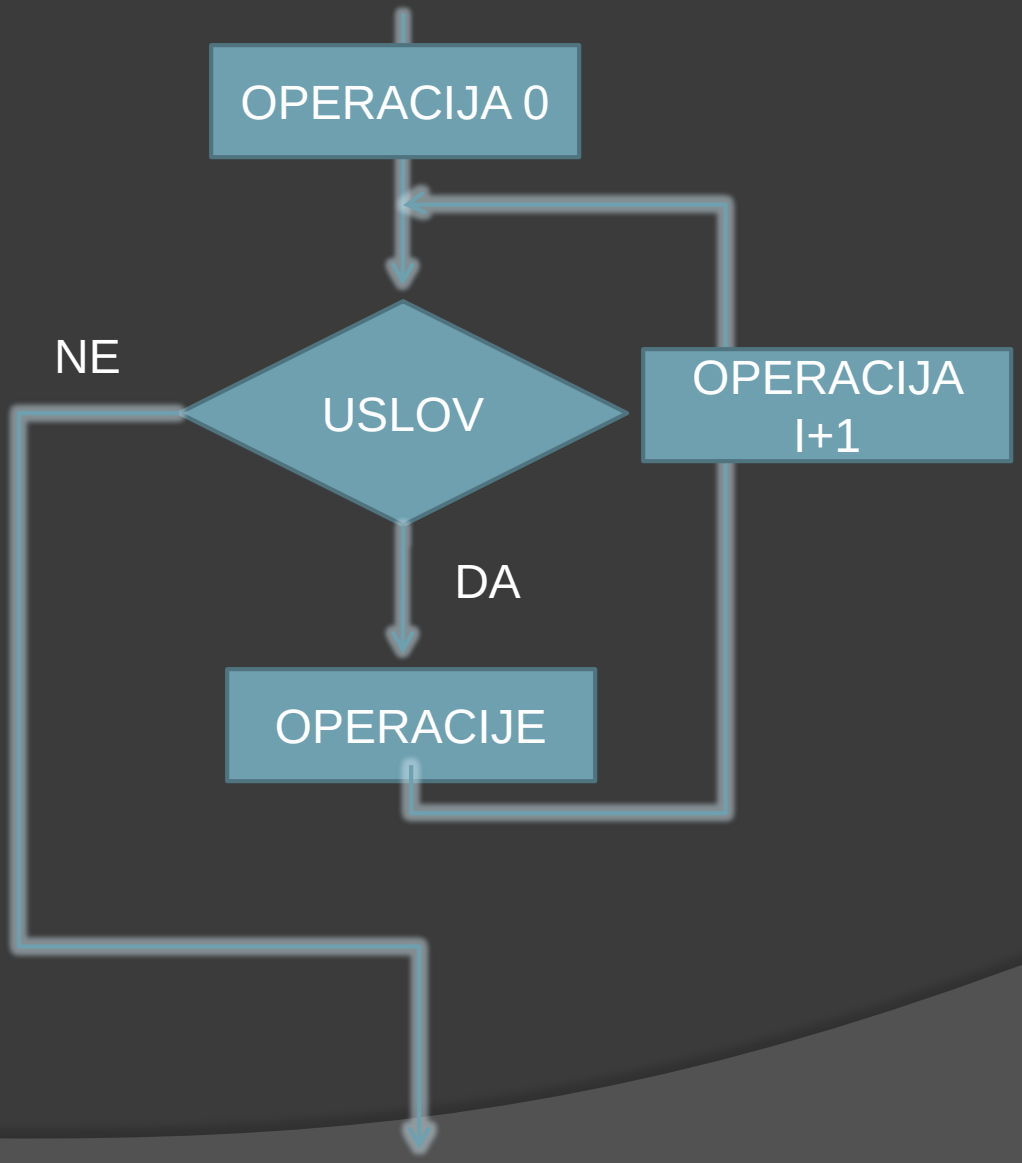
ITERACIJA

- Iteracija omogućava ponavljanje operacija tela iteracije potreban broj puta.
- Iteracije sa izlaskom na vrhu
  - WHILE ulsov DO operacije tela iteracije
  - FOR uslovi i naredbe, operacije tela iteracije
- Iteracije sa izlaskom na dnu
  - DO operacije tela iteracije WHILE uslov
  - REPEAT operacije tela iteracije UNTIL uslov

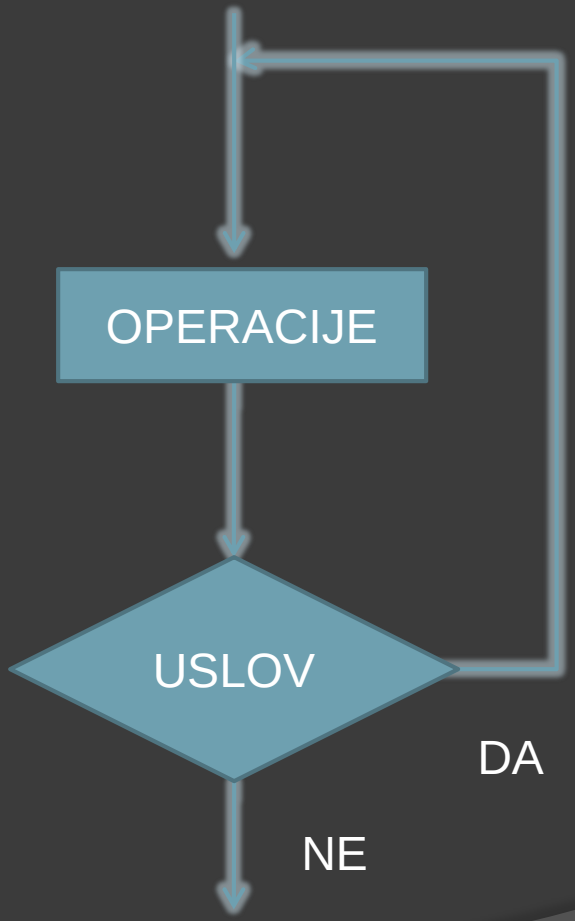
WHILE-DO



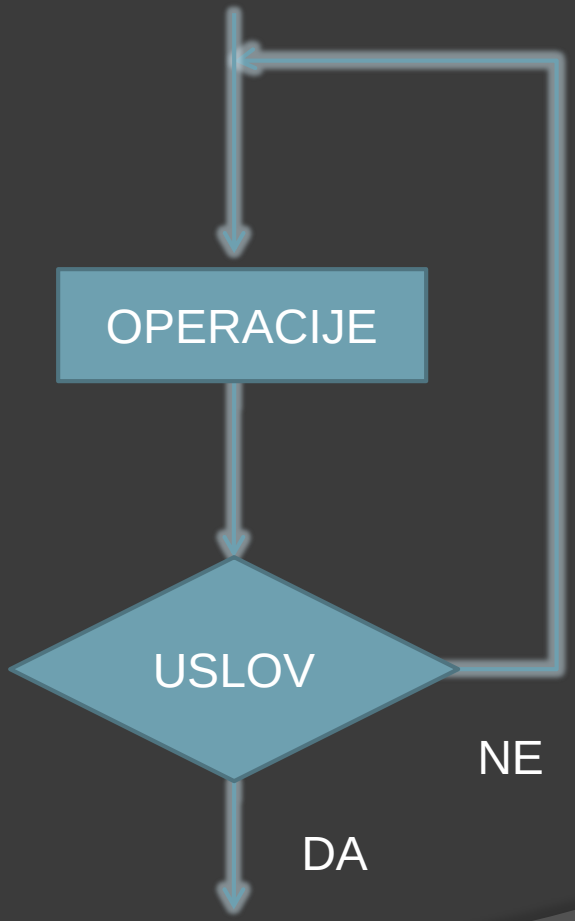
FOR



DO-WHILE



REPEAT-UNTIL



PSEUDOJEZIK

- Specifikacija algoritma može da bude izražena i pseudojezikom
- Pseudojezik koji ćemo koristi je blizak programskom jeziku Pascal.

Konvencije pseudojezika

- Sadži uobičajene kontrolne strukture ciklusa (**while,repeat i for**), osnovne i višestruke selekcije. Pored toga postoji i beskonačna petlja **loop**.
- Telo kontrolne strukture je identirano, a kraj je kontrolne strukture je označen izvodnom reči(end + ime strukture) npr. **end_while.**
- Pored obične dodele a=e vrednosti postoji i višestruka dodela a=b=e (b=e,a=e).
- Postoji naredba razmene vrednosti dve promenljive a ↔ b (t=b,b=a,a=t).

Konvencije pseudojezika

- Promenljive su najčešće lokalne za datu funkciju ili proceduru, ili globalne ako se eksplicitno naglasi.
- Selekcija elemenata niza se ostvaruje navođenjem indeksa u uglastim zagradama iza imena niza A[i]. Simbol ”..” se koristi za označavanje osega elemenata niza. (A[1...i] označava A[1],..., A[i])
- Pristup polju zapisa ukazanog pokazivačem se vrši navođenjem imena polja iza kojeg ide ime pokazivača u malim zagradama. Na primer , ako jedan objekat ima polja x i y, a na njega ukazuje pokazivač p, ovim poljima se pristupa sa x(p) i y(p).
- Pristup polju zapisa zadatog imenom se ostvaruje navođenjem imena zapisa i tačke iza čega ide ime polja. Na primer, polje x zapisa z se referencira kao z.x

Konvencije pseudojezika

- Mehanizam prenosa parametara u potprogram je po vrednosti i po referenci, mada to nije sintaksno posebno naglašeno. Ako je potprogram funkcija, ona takođe vraća vrednost dobijenu izračunavanjem izraza u naredbi return.
- Rekurzija je dozvoljena
- U nekim slučajevima pseudo kod ima neformalne jezičke konstrukcije u engleskom jeziku, čije je značenje očigledno iz prevoda.
- Velikim slovima su označeni pozivi ranije definisanih funkcija ili procedura.
- Komentari nisu uključeni jer su algoritmi podrobno objašnjeni u pratećem tekstu