

LOGIKA PROGRAMIRANJA

Svaki informacijski sustav se sastoji od:

- baze podataka i
- skupa programa.

Najjednostavniji način (metoda) definiranja logike programa je tekstualni opis.

Najčešći način je crtanje dijagrama toka programa ili poznatije kao crtanje FLOWCHART-a.

Metode za definiranje logike rada programa:

- dijagram toka programa ,
- stablo odlučivanja,
- Nassi-Schneidermanov dijagram,
- pseudokod,
- tablice odlučivanja,
- Warnier-Orr-ov dijagram,
- dijagram programskog proizvoda,
- strukturna karta,
- dijagram toka podataka i dr.

Prikaz algoritma

TEKSTUALNI

- prirodni jezik – narativni opis algoritma

GRAFIČKI

- Dijagram toka (Blok dijagram (engl. flowchart))
- slikovno, grafičko predstavljanje algoritma

PSEUDOKOD

– koristi se tekstualni oblik prikaza algoritamskih koraka s formaliziranim programskim strukturama, odgovara redosljedu pojavljivanja programskih izraza pri konkretnoj implementaciji u programskom jeziku

Nassi-Scheidermann diagram - strukturirani dijagrami toka ili iteracijski dijagrami, svaki dijagram odgovara jednoj kontrolnoj programskoj strukturi: sekvenciji, selekciji ili iteraciji

Programski jezik – algoritam se u računalu implementira pomoću konkretnog programskog jezika

Iskustvo pokazuje da je najteži dio rješavanja programerskog problema na računalu upravo razvijanje algoritma za rješenje.

Kad je jednom određen ispravan algoritam , razvoj programa za sam algoritam prilično je jednostavan.

ALGORITAM I PROGRAM

Algoritam: Opis poslova i redsljed izvođenja koji će nakon konačnog broja ponavljanja dovesti do suvislog rezultata

Primjer

*Zadati (učitati) podatak N
Postaviti Nf=1
Za svaki i=1 do N pomnožiti Nf s i
Ispisati rezultat Nf*

Program: Niz poslova koji se izvode određenim redosljedom. U računalu je: program u memoriji, procesor automatski izvodi program na "datoj adresi" memorije.

```
scanf ("%d",&N);  
Nf=1;  
for (i=1;i<=N;i++) Nf *= i;  
printf ("\nRezultat je %d",Nf);
```

Prikaz algoritma, pseudo-kod

Rečenice pisane u govornom jeziku ali u obliku "naredbi"

Tako da su "slične" naredbama u programskom jeziku

```
Postaviti Nf=1  
Zadati (učitati) podatak N  
Ako je N>=0 onda  
    Za svaki i=1 do N  
        pomnožiti Nf s i  
    Ispisati rezultat Nf  
    U suprotnom  
        Ispisati poruku o pogreški  
Kraj selekcije
```

Sekvencija
Selekcija
Iteracija

Primjer: Najveći broj

Zadano je:

Skup podataka u datoteci:
PODACI.DAT

23.04

2.005

12.45

-8.224

30.01

18.94

Treba pronaći
najveći broj iz
datog skupa
podataka!

- nije poznato koliko podataka ima
- podacima se pristupa sekvencijalno

Primjer: Najveći broj

- plan rješavanja (algoritam)

- otvoriti datoteku PODACI.DAT - fopen
- pročitati prvi podatak iz datoteke i postaviti ga za **xmax** - fscanf xmax
- Ponavljati (do kraja podataka) - fscanf
 - čitati iz datoteke u varijablu **x**
 - ako je **x** veći od **xmax** onda - if (x>xmax)
 - postaviti **xmax = x**
- ispisati rješenje **xmax** na ekran - printf xmax

Primjer 2: Najveći broj i koliko ih ima (algoritam Nxmax)

Rješenje:

Učitati xmax

Postaviti nmax=1

Sve dok ima podataka činiti

Učitati x

Ako je x=xmax povećati nmax za 1

Ako je x>xmax

Postaviti xmax=x

Postaviti nmax=1

Kraj Ako je..

Kraj Sve dok...

RJEŠENJE NIJE POTPUNO ISPRAVNO !!

Primjer 2: Najveći broj i koliko ih ima (algoritam Nxmax)

Poboljšano rješenje:

Učitati x

Postaviti xmax=x

Postaviti nmax=0

Sve dok ima podataka činiti

Ako je x=xmax povećati nmax za 1

Ako je x>xmax

Postaviti xmax=x

Postaviti nmax=1

Kraj Ako je..

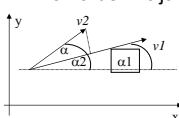
Učitati x

Kraj Sve dok...

Algoritmi s vektorima i matricama (1)

Skalarni produkt dva vektora

Fizikalna definicija:



$$v1 \cdot v2 = v1 \cdot v2 \cdot \cos(\alpha)$$

$$\alpha = \alpha_2 - \alpha_1$$

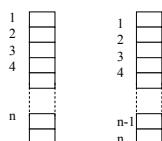
$$v1 \cdot v2 \cdot \cos(\alpha) = v1 \cdot v2 \cdot \cos(\alpha_2 - \alpha_1) =$$

$$v1 \cdot \cos(\alpha_1) \cdot v2 \cdot \cos(\alpha_2) +$$

$$v1 \cdot \sin(\alpha_1) \cdot v2 \cdot \sin(\alpha_2) =$$

$$v1_x \cdot v2_x + v1_y \cdot v2_y$$

Matematička definicija: $Sp = \sum v1_i \cdot v2_i$



Algoritam:

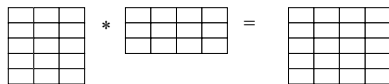
Postaviti Sp=0
Za svaki i=1 do n povećati
Sp za $v1[i] \cdot v2[i]$

Algoritmi s vektorima i matricama (2)

Primjer: Naći srednju vrijednost uz uvjet $V[i] > x$

Postaviti S=0
Postaviti m=0
Za svaki i=1 do n činiti
Ako je $V[i] > x$ činiti
Povećati S za $V[i]$
Povećati m za 1
Kraj Ako je ...
Kraj Za svaki i
Podijeliti S s m
Ispisati S

Algoritmi s vektorima i matricama (3) Množenje matrica



A B C

Definicija: $c_{ij} = \sum_{l=1}^n a_{il} \cdot b_{lj}$

Za svaki i=1 do m

Za svaki j=1 do k činiti

Postaviti $C[i][j]=0$

Za svaki l=1 do n

povećati $C[i][j]$ za $A[i][l]*B[l][j]$

Kraj za svaki j

Pravljenje dijagrama toka

Definicija

Dijagrami toka - grafičko predstavljanje redoslijeda svih radnji - operacija, kretanja, inspekcije (odobrenja), kašnjenja, odluka i zaliha - koje su sastavni dio nekog procesa.

- koriste se simboli za predstavljanje raznih operacija i/ili procesa koji se obavljaju.
- standardni simboli - lakše tumačenje dijagrama, mada standardiziranje simbola nema isti značaj kao redoslijed aktivnosti koje čine proces.

Namjena dijagrama toka

- Dokumentiraju proces i međusobni odnos koraka u procesu.
- Prikazuju stvarnu i idealnu putanju toka proizvoda ili procesa.
- Pokazuje probleme i moguća poboljšanja.
- Mogu se raditi za cjelokupne proizvodne linije, sa svim komponentama, jednom osobom ili jednom komponentom u procesu, kombinacijom ljudi i strojeva, za razne oblike toka transakcija i dokumentacije, itd.

Važna područja koja se prate dijagramima toka

- Radno intenzivni procesi
- Kašnjenja
- Ciklusi kontrole
- Dokumentacija
- Dupliranja
- Propusti

Osnovni tipovi dijagrama toka

- Funkcijski dijagrami -
- Dijagrami procesa -
- Dijagrami za opisivanje toka procesa -


Osnovni simboli


najkorisniji osnovni simboli u industrijskom inženjerstvu i pisanju procesa




Osnovni simboli

najkorisniji osnovni simboli u industrijskom inženjerstvu i pisanju procesa


Kašnjenje - Kašnjenje se obično prikazuje polukrugom ili velikim slovom D. Do kašnjenja na nekom objektu dolazi kada uvjeti, osim kada namjerno izmjene fizičkih ili kemijskih osobina objekta ne zahtijevaju da se odmah poduzme slijedeći planirani korak.



Skladištenje - Do skladištenja dolazi kada se neki objekt čuva i štiti od neovlaštenog uklanjanja. Ujedi simbol obično pokazuje ulaznu robu. Kada se trokut okrene, on obično predstavlja skladištenje ili punjenje.



Transport - Objekt se transportira kada se premješta sa jednog mjesta na drugo, sa izuzetkom kada je takvo premještanje dio operacije ili im je uzrok operacija na radnoj stanici, tokom operacije i/ili pregleda/verifikacije/odobrenja.


Operacije i inspekcija (ili kombinirana aktivnost) - Kada se dvije aktivnosti obavljaju na istoj radnoj stanici ili paralelno, simboli se mogu kombinirati. Obično se kombiniraju poslovi operacije i inspekcije.

Osnovni simboli

najkorisniji osnovni simboli u industrijskom inženjerstvu i pisanju procesa


Konektor - Ovaj simbol pokazuje nastavak dijagrama toka sa jedne stranice na drugoj ili od romba odlučivanja na drugoj stranici ili u drugom procesu.


U složenijim dijagramima toka se koristi još jedan simbol - **Konektor "van stranice"**. Kod njega je princip označavanja isti kao kod prethodno opisanog konektora, ali se u ovom slučaju konektorima dodjeljuju imena čvorova (na primjer, čvor A1 je gdje se završava jedan segment toka, a drugi čvor A1 bio bi gdje drugi tok prikazan na drugom mjestu u dijagramu toka počinje). Tada se koristi konektor "van stranice" da pokaže da je tok na drugoj stranici.

Terminologija dijagrama toka procesa

- **Aktivnost** - Glavni procesi ili komponente unutar funkcije.
Analiza - Ispitivanje i tumačenje podataka prikazanih dijagramom toka kako bi se uklonile nepotrebne i otkrile bolje metode.
Kašnjenje - Simbol koji se koristi da bi se pokazalo da je za posao koji se obavlja uvijek potrebno više vremena nego što je predviđeno, u odnosu na vrijeme koje je potrebno da se isti posao obavi u idealnim uvjetima (odnosno, razlika između "očekivanog vremena" i "proteklog vremena").

Terminologija dijagrama toka procesa

- Funkcija** - Glavne odgovornosti u organizaciji. Funkcije se sastoje od aktivnosti.
- Funkcijski dijagram toka** - da bi se opisala međusobna interakcija aktivnosti.
- Operacija** - Operacija na procesu koji se odvija, uključuje prenošenje, pregledanje ili skladištenje.
- Proces** - Niz zadataka koji se obavljaju jedan za drugim, da bi proizveli proizvod.
- Analiza toka procesa** - opisivanje procesa riječima i simbolima, za tumačenje i proučavanje opisa radi poboljšanja procesa.

Terminologija dijagrama toka procesa

- Dijagram toka procesa** - Dijagram za opisivanje redoslijeda i međusobnog odnosa poslova koji čine jednu aktivnost.
- Simboli toka procesa** - Simboli koji se koriste prilikom ucrtavanja toka procesa u dijagramu toka
- Skladištenje** - Simbol za skladištenje koji pokazuje da je element uskladišten ili da je podatak unesen.
- Posao** - Osnovni koraci koji su neophodni za obavljanje neke aktivnosti.
- Transport** - Prenosjenje objekta sa jednog mjesta na drugo, osim ako je sastavni dio operacije ili je prouzrokovan operacijom na radnom mjestu tokom pregledanja/verifikacije/odobravanja operacije.

Analiza dijagrama toka

- Funkcijski dijagrami
- Dijagrami toka procesa
- Dijagrami opisa toka procesa
- Sumarni

- **Pregled**

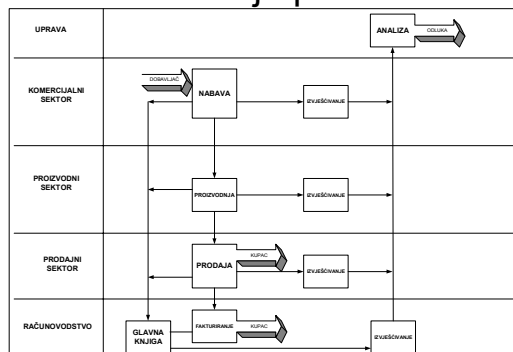
Kada se dovrši analiza dijagrama, napravite pregled inicijalnih procesa, napravite pregled prijedloga promjena i usporedite ta dva pregleda da biste ustanovili moguće poboljšanje rezultata.

- **Softver**

Postoji niz softverskih paketa pomoću kojih se mogu izrađivati dijagrami toka. Onaj koji se najviše koristi u areni MS Officea je Visio. Međutim, vrijedi napomenuti da je ovaj program znatno moćniji nego što je to potrebno za crtanje dijagrama toka.

SmartDraw je također sasvim dobar. On nema tako brojne šablone simbola kao Visio, ali se sasvim dobro pokazuje za većinu aplikacija dijagrama toka i organizacijskih dijagrama. I jeftiniji je.

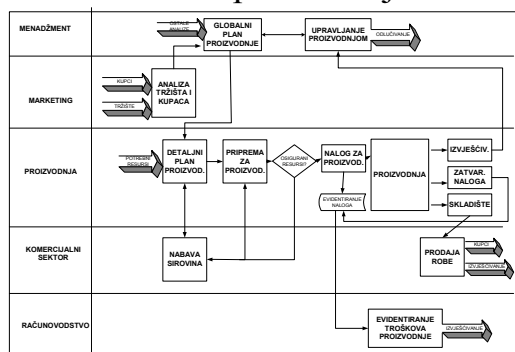
Modeliranje procesa



Proces: proizvodnja

- Primarna svrha ovog procesa jeste uspostava informacijskih tokova za potporu:
 - izvršavanje poslova vezanih za proizvodnju robe, osiguranje sirovina potrebnih za proizvodnju, praćenje troškova proizvodnje
 - donošenja odluka vezano za procese proizvodnje, planiranje proizvodnih kapaciteta, upravljanje troškovima proizvodnje
 - pripremanje internih i eksternih izvješća

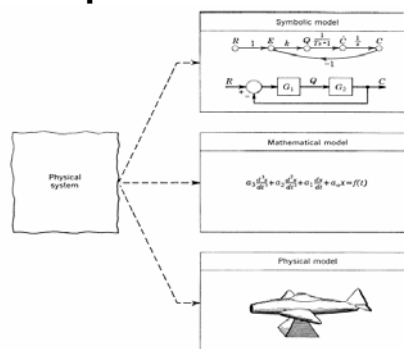
Proces: proizvodnja



Definiranje problema

- Uočava se problem.
 - Određuje način rješavanja.
 - Analizira postojeće stanje.
 - Analiziraju iskustva u radu na ovakvim i sličnim zadacima.
 - Odabiru metode rada.
- U ovoj fazi se još definira fizikalni i matematički model sustava ili procesa čije ponašanje se implementira na računalu.

Zrakoplov kao fizikalni sustav



Fizikalni model predstavlja uvijek model realnog sustava ili procesa koje se odvija u prirodi

Matematički model je skup matematičkih postupaka i relacija kao i putova njihovog rješavanja, koji moraju egzaktno determinirati postavljenu fizikalni model.

Algoritam predstavlja točno propisani postupak za izvršavanje, određenim redoslijedom, definiranih postupaka koji vode do rješenja postavljenog zadatka.

Algoritam se dizajnira primjenom dijagrama toka i pseudokoda.

Standard crtanja **dijagrama toka programa** izrađen na osnovi preporuke R 1028 Međunarodne organizacije za standardizaciju od ožujka 1969. godine.

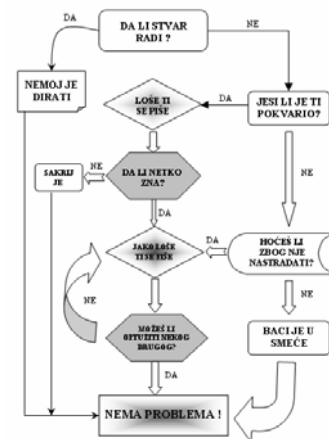
Simboli propisani tim standardom služe, u sustavima za obradu informacija, za prikaz:

- protoka podataka i dokumenata i
- odvijanje operacija programa.

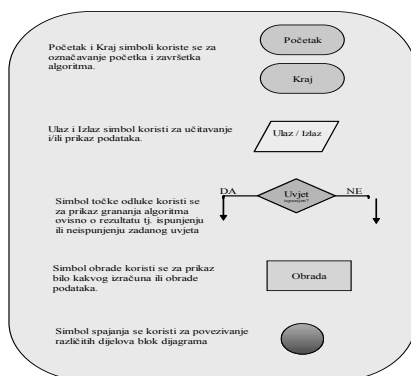
Dijagrami protoka podataka sastoje se :

- simbola za obradu podataka,
- simbola za nosače podataka i
- simbola linija protoka podataka i dokumenata.

Dijagrami odvijanja operacija programa opisuju odvijanje operacija u nekom sustavu obrade informacija, ovisno o trenutno raspoloživim podacima.



Simboli dijagrama toka

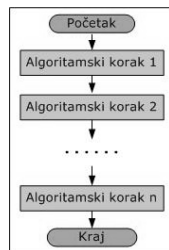


Logičke strukture (Bohm i Jacopini)

- **LINIJSKA (SEKVENCIJA)** – koraci slijede jedan drugog u logičkom slijedu
- **RAZGRANATA (SELEKCIJA (GRANANJE))** – ovisno o određenom uvjetu postoje alternativni koraci
- **CIKLIČKA (ITERACIJA (PONAVLJANJE))** – neki se koraci ponavljaju dok je ispunjen određeni uvjet

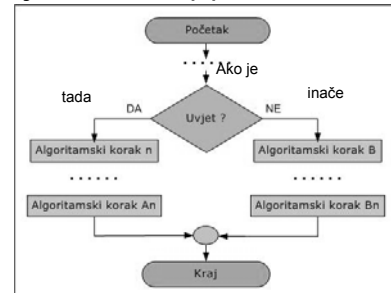
Strukturirano programiranje

Sekvencija – slijed naredbi koje se izvršavaju jedna za drugom



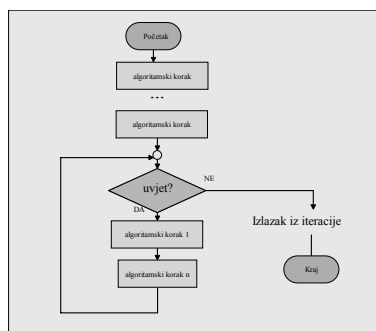
Strukturirano programiranje

Selekcija – uvjetno grananje (Ako je-tada-inače) (IF-THEN-ELSE)



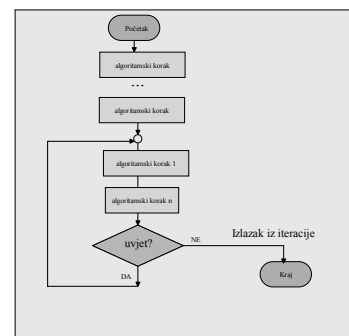
Strukturirano programiranje

Iteracija – “DO WHILE”



Strukturirano programiranje

Iteracija – “DO UNTIL”



Vrijednost nasuprot varijabla

Logička razlika:

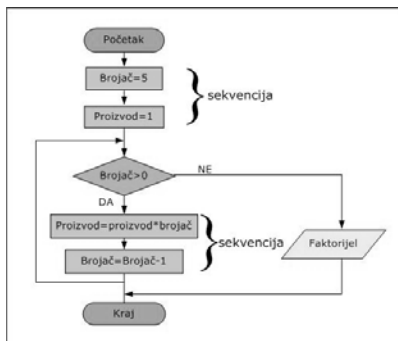
- Vrijednost je “pojedinačna konstanta” (npr. 5), vrijednost nema lokaciju u vremenu ili prostoru. Nema ažuriranja!! Vrijednost se može predstaviti u memoriji pomoću kodiranja, i takvo predstavljanje tj. pojava ima lokaciju u vremenu i prostoru.
- Varijabla služi za pohranu predstave tj. pojave vrijednosti. Varijabla ima lokaciju u vremenu i prostoru. Može se ažurirati !!!
- predstavljanje vrijednosti je “model” koncept
- Kodiranje predstavljanja je “implementacijski” koncept

Primjeri blok dijagrama

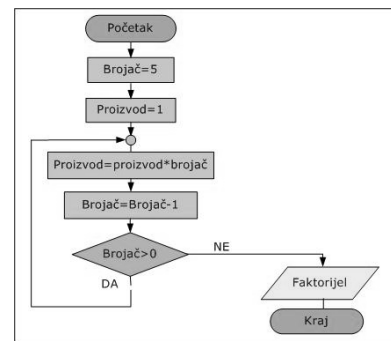
Primjer: Algoritam za izračunavanje faktoriјela broja 5 – 1x2x3x4x5

1. Postaviti brojač na 5
2. Postaviti proizvod na 1
3. Sve dok je brojač veći od nule ponavljati slijedeće korake:
 - 3.1. pomnožiti proizvod i brojač
 - 3.2. umanjiti brojač za 1
4. Kada vrijednost brojača postane nula, zaustavlja se izvršavanje algoritamskih koraka.

Primjeri blok dijagrama



Primjeri blok dijagrama



Koncept slijeda u programiranju

- **Ulaz** – prvo se unose podaci
- **Obrada** – kada su podaci unešeni, mogu se obraditi kako bi dobili informacije
- **Izlaz** – informacije mogu naposljetku biti prikazane

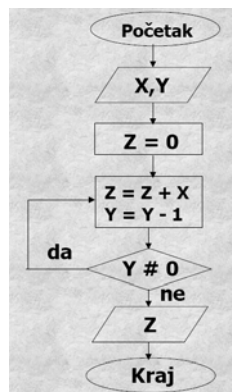
Zaključci o slijedu

- Kod rješavanja bilo kojeg problema nužno je odlučiti da li pojedini koraci dolaze prije ili poslije drugih koraka.
- Često efikasnost rješenja ovisi o ispravnosti ove odluke.
- Takve se posljedice ne moraju pojaviti kada redoslijed koraka nije bitan.
- Dakle, zaključno, uočimo i promislimo za koje korake je bitan redoslijed a za koje nije.

Primjer:

Sastaviti algoritam za množenje dvaju proizvoljno zadanih prirodnih brojeva koristeći operaciju zbrajanja.

Rješenje: Neka su ulazne veličine prirodni brojevi x i y s vrijednostima kako slijedi : $x = 14$ i $y = 3$.



Varijable i imena varijabli

- U ovom primjeru, tri će vrijednosti biti pohranjene u računalnoj memoriji – dva množitelja i njihov umnožak.
- Ove vrijednosti nazivamo varijable jer to mogu biti bilo koji brojevi.
- Svakoj varijabli u algoritmu moramo dati ime (variable name) ili (identifier name).
- O imenu odlučujemo prije izrade algoritma, i navodimo ga u rječniku podataka (data dictionary).

Rječnik podataka (Data Dictionary)

- *Ime varijable (Variable Name)* – skraćeno, smisljeno ime za opis entiteta (pojma);
- *Opis (Description)* – kratak opis što varijabla predstavlja (znači);
- *Tip (Type)* – uobičajeno je: cjelobrojan (integer), realni (real), slovni (char), tekstualni (string) ili logički (boolean).

Rječnik podataka za primjer

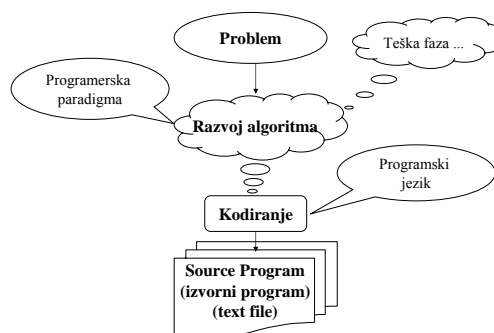
Ime	Tip	Opis
broj1	integer	Prvi uneseni broj
broj2	integer	Drugi uneseni broj
prod	integer	Umnožak unesenih brojeva



Napomena o algoritmima

- Slijedi posebno važna napomena uz algoritme.
- *Algoritam se uvijek izrađuje neovisno o programskom jeziku.*
- Drugim riječima, ako je algoritam ispravno načinjen, mora se moći na osnovu njega napraviti program u bilo kojem programskom jeziku.
- Zbog toga se algoritmi pišu na engleskom (hrvatskom) i koriste se uobičajeni simboli i pojmovi.

Rješenje problema



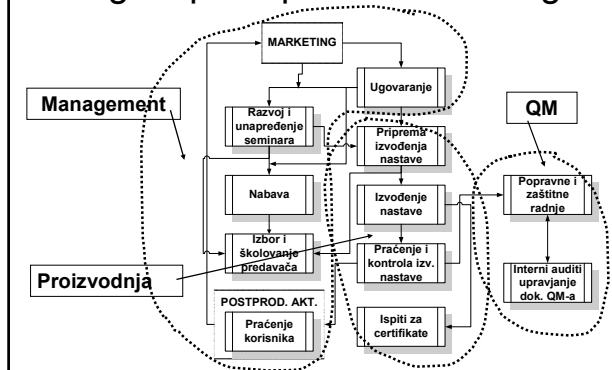
Postupci u IT treningu/1

- Management
 - Ugovaranje
 - Razvoj novih i unapr. postojećih seminara
 - Nabava
 - Izbor i školovanje predavača
 - Praćenje korisnika
- Proizvodnja
 - Priprema izvođenja nastave
 - Izvođenje nastave
 - Praćenje i kontrola izvođenja nastave
 - Ispiti za certifikate

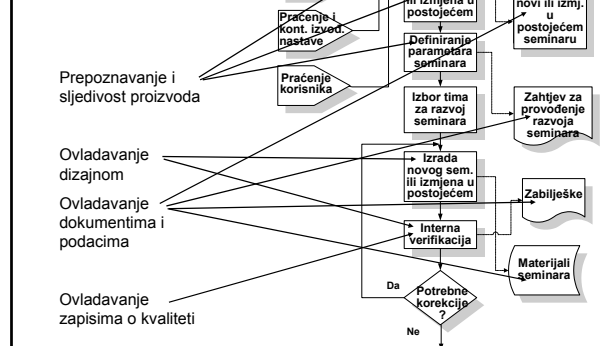
Postupci u IT treningu/2

- QM
 - Interni auditi
 - Upravljanje dokumentima i podacima
 - Upravljanje zapisima kvalitete
 - Popravne i zaštitne radnje

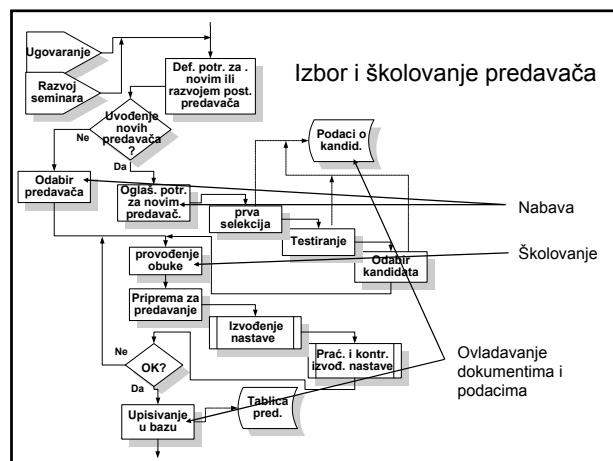
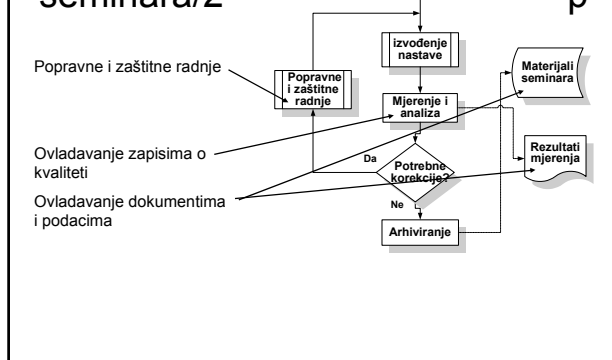
Pregled postupaka u IT treningu



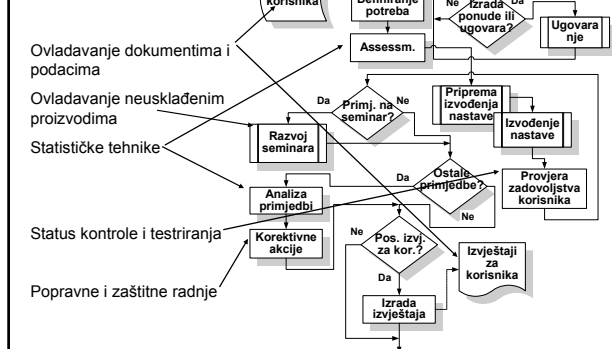
Razvoj seminara/1



Razvoj seminara/2



Praćenje korisnika



Priprema izvođenja nastave

