

OBRAZOVNI SOFTVER

Stefan Mišković

Matematički fakultet, Beograd

Sadržaj

1	Uvod u LaTeX	3
1.1	Osnovne osobine LaTeX-a	3
1.2	Priprema radnog okruženja	3
1.3	Struktura LaTeX dokumenta	4
1.4	Paketi i osnovna podešavanja	6
1.5	Jezik i kodiranje	7
1.6	Organizacija i formatiranje teksta	8
1.7	Izgled stranice	11
2	Matematički sadržaj u LaTeX-u	14
2.1	Matematički režim	14
2.2	Matematički simboli i izrazi	15
2.3	Jednačine i višeredni izrazi	19
2.4	Matrice i sistemi jednačina	21
2.5	Matematički fontovi	23
2.6	Definicije, teoreme i dokazi	24
3	Dodatni elementi LaTeX dokumenta	27
3.1	Slike	27
3.2	Tabele	28
3.3	Oznake i reference	29
3.4	Hiperveze	31
3.5	Literatura i citiranje	32
4	Grafički prikazi u TikZ-u	34
4.1	Osnovna struktura crteža	34
4.2	Koordinate, tačke i linije	35
4.3	Geometrijski oblici i putanje	37
4.4	Tekst i matematičke oznake	39
4.5	Stilovi i oblikovanje crteža	41
4.6	Složeniji geometrijski crteži	44
5	Prezentacije u Beamer-u	47
5.1	Osnovna struktura prezentacije	47
5.2	Slajdovi i organizacija sadržaja	48
5.3	Tekst, liste i blokovi	51
5.4	Matematički i grafički sadržaj	53
5.5	Postepeno prikazivanje sadržaja	55
6	Veb-tehnologije	58
6.1	Bootstrap	58
6.2	MathJax	58
6.3	GeoGebra	60

7	Projekat iz Obrazovnog softvera	62
7.1	Cilj projekta	62
7.2	Prijava teme i bodovanje	62
7.3	Celine projekta	62
7.4	Predaja i odbrana	62
7.5	Primer projekta	63

1 Uvod u LaTeX

1.1 Osnovne osobine LaTeX-a

LaTeX je sistem za pripremu i oblikovanje dokumenata. Posebno je pogodan za izradu naučnih, stručnih i tehničkih tekstova, a naročito dokumenata koji sadrže matematičke izraze. Omogućava organizovanje teksta u celine, automatsko numerisanje različitih elemenata, povezivanje referenci, izradu sadržaja i literature i druge postupke koji su česti pri pripremi složenijih dokumenata.

Rad u LaTeX-u razlikuje se od rada u klasičnim programima za obradu teksta. U takvim programima korisnik neposredno uređuje izgled dokumenta i tokom rada vidi približno njegov konačni oblik. Ovakav način rada često se označava skraćenicom WYSIWYG, od engleskog izraza *what you see is what you get*.

Kod LaTeX-a dokument se najpre zapisuje u tekstualnom izvornom fajlu. Takav fajl obično ima nastavak `.tex` i sadrži tekst dokumenta zajedno sa komandama koje opisuju njegovu strukturu i način prikazivanja pojedinih elemenata. Izvorni fajl se zatim kompajlira, pri čemu se dobija konačni dokument, najčešće u PDF formatu.

Važna osobina ovakvog pristupa jeste razdvajanje sadržaja i strukture dokumenta od njegovog konačnog izgleda. Na primer, umesto da korisnik za svaki naslov ručno određuje veličinu slova, razmak i druge osobine, u LaTeX dokumentu navodi se da određeni tekst predstavlja naslov sekcije. LaTeX zatim, na osnovu izabrane vrste dokumenta i podešavanja, određuje kako će svi naslovi tog nivoa izgledati. Na taj način lakše se postiže ujednačen izgled celog dokumenta.

Sličan princip koristi se i za druge elemente dokumenta. Matematički izrazi zapisuju se posebnim komandama, slike i tabele mogu se automatski numerisati, a na numerisane elemente moguće je pozivati se iz drugih delova teksta. Ako se njihov položaj ili redosled promeni, LaTeX može automatski prilagoditi numeraciju i odgovarajuće reference. Zbog toga je posebno pogodan za veće i dobro strukturirane dokumente.

Izvorni LaTeX dokument sastoji se od običnog teksta i posebnih komandi i okruženja. Komande se uobičajeno prepoznaju po znaku `\` kojim počinju, dok se okruženja koriste kada određeni deo dokumenta treba posmatrati kao jednu celinu. Njihovu sintaksu i ulogu detaljnije ćemo razmotriti u narednim podsekcijama.

Jedan veoma jednostavan LaTeX dokument može izgledati ovako:

```
\documentclass{article}

\begin{document}

Ovo je prvi LaTeX dokument.

\end{document}
```

Kompajliranjem ovog izvornog fajla dobija se dokument koji sadrži jednu rečenicu. Sam izvor, međutim, pored te rečenice sadrži i komande koje određuju vrstu dokumenta i označavaju početak i kraj njegovog sadržaja. U narednim podsekcijama detaljnije ćemo razmotriti osnovne delove LaTeX dokumenta i pravila po kojima se oni zapisuju.

1.2 Priprema radnog okruženja

LaTeX dokument piše se u tekstualnom izvornom fajlu, ali je za njegovo kompajliranje potrebno odgovarajuće programsko okruženje. Pri lokalnom radu ono se najčešće sastoji od

LaTeX distribucije i editora u kome se izvorni fajl piše i uređuje.

LaTeX distribucija predstavlja skup programa, paketa i drugih komponenti potrebnih za obradu LaTeX dokumenata. Ona omogućava da se izvorni `.tex` fajl kompajlira i da se na osnovu njega dobije konačni dokument. Među najpoznatijim distribucijama nalaze se TeX Live i MiKTeX. Na operativnom sistemu macOS često se koristi i MacTeX, koji uključuje TeX Live i dodatne komponente prilagođene macOS-u.

Pored distribucije, za rad je pogodno koristiti editor namenjen pisanju LaTeX dokumenata. Iako se `.tex` fajl može uređivati u bilo kom programu za rad sa običnim tekstom, specijalizovani editori olakšavaju pisanje koda, ističu njegovu sintaksu, omogućavaju jednostavno pokretanje kompajliranja i često prikazuju dobijeni PDF dokument u okviru istog programa. Jedan od takvih editora je TeXstudio.

Važno je razlikovati editor od LaTeX distribucije. Editor služi za pisanje i uređivanje izvornog dokumenta i omogućava lakše pokretanje odgovarajućih programa, dok distribucija sadrži programe i pakete koji stvarno obrađuju LaTeX kod. Zbog toga instaliranje samog editora nije dovoljno za lokalno kompajliranje LaTeX dokumenata.

LaTeX se može koristiti i bez lokalne instalacije. Overleaf je veb-okruženje u kome se LaTeX dokumenti pišu i kompajliraju neposredno u pregledaču. Korisniku su na jednom mestu dostupni editor izvornog koda i prikaz dobijenog dokumenta, dok se samo kompajliranje obavlja na udaljenom sistemu. Ovakav način rada posebno je pogodan kada želimo da brzo započnemo rad ili da isti dokument uređuje više korisnika.

Bez obzira na izabrano okruženje, osnovni postupak rada je isti. Najpre se napiše ili izmeni izvorni `.tex` fajl, zatim se on kompajlira i pregleda se dobijeni dokument. Ako rezultat nije odgovarajući ili se pri kompajliranju pojavi greška, izvorni fajl se ispravlja i postupak se ponavlja.

Pošto pripremljeno radno okruženje omogućava pisanje i kompajliranje LaTeX dokumenata, u nastavku ćemo razmotriti od kojih se delova jedan takav dokument sastoji i kako se oni zapisuju.

1.3 Struktura LaTeX dokumenta

LaTeX dokument ima određenu osnovnu strukturu koja se ponavlja bez obzira na njegov sadržaj. Na početku se navodi klasa dokumenta, zatim mogu da slede različita podešavanja, dok se sam sadržaj dokumenta zapisuje između komandi koje označavaju njegov početak i kraj.

Jedan jednostavan dokument može izgledati ovako:

```
\documentclass[12pt,a4paper]{article}

% Ovde se nalazi preambula dokumenta.

\begin{document}

Ovo je tekst dokumenta.

\end{document}
```

Prvom komandom određuje se klasa dokumenta. U prikazanom primeru koristi se klasa `article`:

```
\documentclass[12pt,a4paper]{article}
```

Naziv klase navodi se između vitičastih zagrada. Klasa određuje osnovnu organizaciju i izgled dokumenta. Pored klase `article`, postoje i druge klase, kao što su `report` i `book`, namenjene drugačije organizovanim dokumentima. Posebnu klasu `beamer`, koja se koristi za izradu prezentacija, razmotrićemo kasnije.

Između uglastih zagrada mogu se navesti dodatne opcije klase. U ovom primeru opcija `12pt` određuje osnovnu veličinu slova, dok `a4paper` određuje format papira. Opcije nisu obavezne, pa je ispravan i jednostavniji zapis

```
\documentclass{article}
```

Deo izvornog fajla između komande `\documentclass` i komande `\begin{document}` naziva se preambula. U preambuli se navode različita podešavanja, uključuju paketi i po potrebi definišu dodatne komande. Kod sasvim jednostavnog dokumenta preambula može biti i prazna. Pakete i osnovna podešavanja detaljnije ćemo razmotriti u narednoj podsekciji.

Sadržaj dokumenta nalazi se unutar okruženja `document`:

```
\begin{document}
```

Ovo je tekst dokumenta.

```
\end{document}
```

Komanda `\begin{document}` označava početak sadržaja, dok komanda `\end{document}` označava njegov kraj. Sve što treba da bude deo konačnog dokumenta zapisuje se između ove dve komande.

LaTeX kod sastoji se od običnog teksta i posebnih komandi. Komande počinju znakom `\`, iza koga se nalazi njihov naziv. Neke komande ne zahtevaju dodatne podatke, dok druge imaju jedan ili više argumenata. Obavezni argumenti zapisuju se između vitičastih zagrada. Na primer, u komandi

```
\documentclass{article}
```

tekst `article` predstavlja obavezni argument komande `\documentclass`.

Pored obaveznih argumenata, komanda može imati i opcione argumente. Oni se zapisuju između uglastih zagrada i najčešće služe za dodatno podešavanje njenog ponašanja. Tako u zapisu

```
\documentclass[12pt,a4paper]{article}
```

deo `[12pt,a4paper]` predstavlja opcioni argument, dok je `{article}` obavezni argument.

Za delove dokumenta koji predstavljaju određenu celinu često se koriste okruženja. Okruženje počinje komandom `\begin`, a završava se odgovarajućom komandom `\end`. Njihov opšti oblik je

```
\begin{naziv}
```

Primer teksta.

```
\end{naziv}
```

Naziv okruženja na početku i kraju mora biti isti. Okruženje `document` predstavlja osnovni

primer, a kasnije ćemo upoznati druga okruženja namenjena listama, matematičkim izrazima, tabelama i drugim elementima dokumenta.

U izvornom fajlu mogu se pisati i komentari. Komentar počinje znakom %, a tekst od tog znaka do kraja reda LaTeX zanemaruje prilikom kompajliranja. Komentari zato mogu da se koriste za kratka objašnjenja u izvornom kodu ili za privremeno isključivanje pojedinih delova zapisa.

```
% Ovaj red se ne prikazuje.  
Ovaj red se prikazuje u dokumentu.
```

Na ovaj način i najjednostavniji LaTeX dokument ima jasno određenu strukturu. Na početku se bira klasa dokumenta, zatim sledi preambula sa podešavanjima, a unutar okruženja `document` nalazi se njegov sadržaj. U narednoj podsekciji videćemo kako se mogućnosti LaTeX-a proširuju pomoću paketa i kako se u preambuli zadaju osnovna podešavanja dokumenta.

1.4 Paketi i osnovna podešavanja

Osnovne mogućnosti LaTeX-a mogu se proširivati pomoću paketa. Paket sadrži dodatne komande, okruženja ili podešavanja namenjena određenoj vrsti sadržaja ili načinu oblikovanja dokumenta. Na taj način u dokument se uključuju samo mogućnosti koje su za njega potrebne.

Paketi se učitavaju u preambuli dokumenta pomoću komande `\usepackage`. Na primer,

```
\usepackage{amsmath}
```

uključuje paket `amsmath`, koji proširuje mogućnosti LaTeX-a za zapisivanje i organizovanje matematičkih izraza.

Kao i druge LaTeX komande, komanda `\usepackage` može imati opcionu argument. Na primer,

```
\usepackage[margin=2cm]{geometry}
```

uključuje paket `geometry` i istovremeno mu prosleđuje opciju kojom se margine dokumenta postavljaju na 2 cm. Značenje dostupnih opcija zavisi od konkretnog paketa.

Za različite namene koriste se različiti paketi. Među paketima koji se često pojavljuju u jednostavnim naučnim i stručnim dokumentima nalaze se `geometry`, `amsmath`, `amssymb`, `lmodern` i `indentfirst`. Paket `geometry` omogućava jednostavno podešavanje dimenzija stranice i margina. Paket `amsmath` uvodi dodatne mogućnosti za rad sa matematičkim izrazima, dok `amssymb` obezbeđuje dodatne matematičke simbole. Paket `lmodern` omogućava korišćenje familije fontova Latin Modern, a `indentfirst` uvlači i prvi pasus koji sledi nakon naslova sekcije ili podsekcije.

Paket se uključuje samo ako su njegove mogućnosti potrebne. Na primer, pakete za rad sa slikama, hipervezama ili grafičkim prikazima uvešćemo kasnije, kada budemo obrađivali odgovarajuće elemente dokumenta. Na taj način se u preambuli jasno vidi koje dodatne mogućnosti dokument koristi.

Jedan jednostavan dokument sa nekoliko uključenih paketa može izgledati ovako:

```
\documentclass[12pt,a4paper]{article}  
  
\usepackage[margin=2cm]{geometry}  
\usepackage{amsmath}  
\usepackage{amssymb}  
\usepackage{lmodern}
```

```
\usepackage[indentfirst]

\begin{document}

Ovo je primer dokumenta.

\end{document}
```

Sve komande `\usepackage` nalaze se u preambuli, odnosno pre komande `\begin{document}`. Nakon početka okruženja `document` piše se sadržaj dokumenta. Mogućnosti prethodno uključenih paketa tada su dostupne u njegovom nastavku.

Pored uključivanja paketa, u preambuli se mogu nalaziti i druga podešavanja koja važe za ceo dokument. Njih ćemo uvoditi postepeno, zajedno sa elementima na koje se odnose. U narednoj podsekciji razmotrićemo podešavanja potrebna za pravilno pisanje dokumenta na srpskom jeziku.

1.5 Jezik i kodiranje

Pri pisanju dokumenta na jezicima koji koriste slova izvan osnovnog engleskog alfabeta potrebno je voditi računa o načinu na koji su znakovi zapisani u izvornom fajlu i o jezičkim pravilima koja LaTeX primenjuje na dokument. Ova dva pitanja su povezana, ali imaju različitu ulogu.

Izvorni LaTeX fajl je tekstualni fajl u kome svaki znak mora biti predstavljen na određeni način. Danas se za tu svrhu uobičajeno koristi kodiranje UTF-8, koje omogućava zapis velikog broja pisama i znakova. Savremene verzije LaTeX-a podrazumevano očekuju UTF-8, pa se u izvornom fajlu mogu neposredno pisati i slova srpske latinice kao što su č, ć, š, ž i đ.

U starijim LaTeX dokumentima za izbor UTF-8 kodiranja često se sreće paket `inputenc` i zapis

```
\usepackage[utf8]{inputenc}
```

U savremenim verzijama LaTeX-a UTF-8 je podrazumevano ulazno kodiranje, pa ovaj paket obično nije potrebno posebno uključivati.

Pored kodiranja izvornog fajla postoji i kodiranje fonta, koje određuje način na koji su znakovi predstavljeni u fontovima koje LaTeX koristi. Za dokumente pisane srpskom latinicom pogodno je kodiranje T1, koje se bira pomoću paketa `fontenc`.

```
\usepackage[T1]{fontenc}
```

Ovakvo podešavanje omogućava pravilno korišćenje latiničnih slova sa dijakriticima i pogodno je za većinu evropskih jezika koji koriste latinicu.

Samo pravilno predstavljanje znakova nije dovoljno da bi LaTeX znao na kom jeziku je dokument napisan. Jezik dokumenta podešava se pomoću paketa `babel`. Za srpski jezik pisan latinicom koristi se opcija `serbian`.

```
\usepackage[serbian]{babel}
```

Izbor jezika utiče na jezička pravila koja LaTeX primenjuje prilikom oblikovanja dokumenta. To obuhvata, između ostalog, pravilno rastavljanje reči na kraju reda i nazive pojedinih elemenata koje LaTeX automatski generiše.

Jednostavan dokument pisan srpskom latinicom može izgledati ovako:

```
\documentclass[12pt,a4paper]{article}

\usepackage[T1]{fontenc}
\usepackage[serbian]{babel}
\usepackage{lmodern}

\begin{document}

Ovo je tekst na srpskom jeziku.
Možemo neposredno koristiti slova č, ć, š, ž i đ.

\end{document}
```

Na ovaj način podešavaju se kodiranje fonta i jezička pravila potrebna za dokument na srpskoj latinici.

LaTeX omogućava i pripremu dokumenata na srpskom jeziku pisanih ćirilicom. Za takve dokumente koristi se ćirilično kodiranje fonta T2A i odgovarajuća opcija **serbianc** paketa **babel**. Detaljnija podešavanja različitih pisama i fontova nećemo razmatrati, jer za uobičajeni rad sa dokumentima na srpskoj latinici prethodno opisana podešavanja predstavljaju dovoljnu osnovu.

Nakon što smo podesili osnovne osobine dokumenta i jezik na kome se piše, možemo detaljnije razmotriti način organizovanja i formatiranja samog teksta.

1.6 Organizacija i formatiranje teksta

LaTeX omogućava da se sadržaj dokumenta organizuje pomoću komandi koje određuju ulogu njegovih pojedinih delova. Umesto da se izgled svakog naslova ili pasusa podešava ručno, navodi se kojoj vrsti elementa tekst pripada, a LaTeX zatim određuje njegov izgled.

Osnovni podaci o dokumentu mogu se zadati pomoću komandi `\title`, `\author` i `\date`. Naslov se prikazuje pomoću komande `\maketitle`.

```
\documentclass{article}

\usepackage[T1]{fontenc}
\usepackage[serbian]{babel}
\usepackage{lmodern}

\title{Uvod u teoriju grafova}
\author{Petar Petrović}
\date{2026}

\begin{document}

\maketitle

Tekst dokumenta.

\end{document}
```

Komande kojima se zadaju naslov, autor i datum najčešće se nalaze u preambuli, dok se ko-

manda `\maketitle` koristi u telu dokumenta na mestu na kome naslov treba da se pojavi. Ako datum ne želimo da prikazemo, možemo zadati praznu vrednost pomoću komande `\date`.

Duži dokumenti obično se dele na celine. Kod klase `article` za to se najčešće koriste komande `\section`, `\subsection` i `\subsubsection`.

```
\section{Grafovi}
```

Tekst sekcije.

```
\subsection{Osnovni pojmovi}
```

Tekst podsekcije.

```
\subsubsection{Stepen čvora}
```

Tekst podsekcije nižeg nivoa.

LaTeX automatski određuje izgled naslova i njihovu numeraciju. Zbog toga nije potrebno ručno upisivati brojeve sekcija niti posebno podešavati izgled svakog naslova.

Na osnovu strukture dokumenta može se automatski napraviti njegov sadržaj. Za to se koristi komanda

```
\tableofcontents
```

koja se postavlja na mesto na kome sadržaj treba da se pojavi. Stavke sadržaja formiraju se na osnovu naslova odgovarajućih celina dokumenta.

Tekst između naslova organizuje se u pasuse. Novi pasus dobija se ostavljanjem praznog reda u izvornom fajlu.

Ovo je prvi pasus dokumenta. On može da sadrži
više redova u izvornom fajlu.

Ovo je drugi pasus dokumenta.

Sam prelazak u novi red u izvornom fajlu ne označava novi pasus. LaTeX razmake i prelome redova u običnom tekstu prilagođava rasporedu dokumenta. Ako je potrebno eksplicitno preći u novi red bez započinjanja novog pasusa, može se koristiti komanda `\\`. Ovu komandu ne treba koristiti kao zamenu za pravilno razdvajanje pasusa.

Po potrebi se može započeti i nova stranica. Za to se koristi komanda

```
\newpage
```

koja završava trenutnu stranicu i nastavak dokumenta prebacuje na sledeću.

Izgled pojedinih delova teksta može se menjati odgovarajućim komandama. Za podebljani tekst koristi se `\textbf`, za kurzivni tekst `\textit`, dok se pomoću komande `\texttt` dobija tekst pisan fontom fiksne širine.

Ovo je `\textbf{podebljani tekst}`.

Ovo je `\textit{kurzivni tekst}`.

Ovo je `\texttt{tekst fiksne širine}`.

Dobijeni izgled može se videti neposredno u dokumentu. Ovo je **podebljani tekst**. Ovo je

kurzivni tekst. Ovo je **tekst fiksne širine**.

Komanda `\texttt` posebno je pogodna za zapis naziva komandi, fajlova, promenljivih i drugih delova teksta koji predstavljaju programski kod. Komanda `\textit` neposredno određuje kurzivni izgled teksta i može se koristiti, na primer, pri navođenju stranih termina.

Za naglašavanje određenog dela teksta postoji i komanda `\emph`. Njom se označava da neki deo teksta treba istaći, bez neposrednog zadavanja konkretnog izgleda. U običnom tekstu rezultat njenog korišćenja najčešće je kurziv.

```
Ovaj deo teksta je \emph{posebno važan}.
```

U dokumentu se dobija rečenica: Ovaj deo teksta je *posebno važan*.

Kada je potrebno navesti više povezanih stavki, mogu se koristiti liste. Nenumerisana lista pravi se pomoću okruženja `itemize`, a svaka stavka počinje komandom `\item`.

```
\begin{itemize}
\item prva stavka
\item druga stavka
\item treća stavka
\end{itemize}
```

Ovaj zapis daje nenumerisanu listu:

- prva stavka
- druga stavka
- treća stavka

Numerisana lista pravi se na isti način pomoću okruženja `enumerate`.

```
\begin{enumerate}
\item prvi korak
\item drugi korak
\item treći korak
\end{enumerate}
```

Dobija se lista u kojoj LaTeX automatski određuje numeraciju stavki:

1. prvi korak
2. drugi korak
3. treći korak

Liste se mogu i ugnježdavati kada je potrebno prikazati više nivoa stavki.

Kraće dodatno objašnjenje može se izdvojiti u fusnotu pomoću komande `\footnote`. Tekst fusnote navodi se kao njen obavezni argument.

```
LaTeX se često koristi za pripremu naučnih
dokumenata.\footnote{Posebno je pogodan za
dokumente sa matematičkim sadržajem.}
```

LaTeX automatski dodaje oznaku fusnote u tekst i prikazuje njen sadržaj pri dnu odgovarajuće stranice.

Pojedini znakovi imaju posebno značenje u LaTeX kodu i zato se ne mogu uvek neposredno zapisati kao običan tekst. Među najčešćim takvim znakovima su %, \$, &, #, _, { i }. Kada želimo da se neki od njih prikaže u dokumentu, ispred njega se piše znak \.

Znak \ i sam ima posebno značenje jer njime počinju LaTeX komande. Kada želimo da ga prikažemo kao znak u običnom tekstu, možemo koristiti komandu `\textbackslash`.

Organizovanjem dokumenta u celine i pasuse, odgovarajućim formatiranjem teksta i korišćenjem lista i fusnota može se pripremiti najveći deo običnog tekstualnog sadržaja. U narednoj podsekciji razmotrićemo osnovna podešavanja izgleda stranice, kao što su margine, numeracija stranica, zaglavlja i podnožja.

1.7 Izgled stranice

Klasa dokumenta određuje njegov osnovni izgled, ali se pojedine osobine stranice mogu dodatno podešavati. Među najčešćim podešavanjima nalaze se format papira, margine, numeracija stranica, zaglavlje i podnožje.

Format papira i osnovna veličina slova mogu se zadati već prilikom izbora klase dokumenta. Na primer,

```
\documentclass[12pt,a4paper]{article}
```

određuje da se koristi papir formata A4 i osnovna veličina slova od 12 tačaka.

Za jednostavno podešavanje margina često se koristi paket `geometry`. Ako želimo da sve margine budu široke 2 cm, možemo napisati

```
\usepackage[margin=2cm]{geometry}
```

Pojedine margine mogu se podešavati i zasebno. Na primer,

```
\usepackage[
top=2cm,
bottom=3cm,
left=2cm,
right=2cm
]{geometry}
```

postavlja gornju, levu i desnu marginu na 2 cm, dok donja margina iznosi 3 cm.

LaTeX kod klase `article` podrazumevano numerise stranice dokumenta. Način prikazivanja zaglavlja, podnožja i broja stranice određuje se stilom stranice. Jedan od osnovnih stilova je `plain`.

```
\pagestyle{plain}
```

Kod ovog stila broj stranice prikazuje se u podnožju. Ako ne želimo zaglavlje, podnožje ni broj stranice, možemo koristiti stil `empty`.

```
\pagestyle{empty}
```

Komanda `\pagestyle` utiče na stranice dokumenta u njegovom nastavku. Ako želimo da promenimo stil samo jedne stranice, koristi se komanda `\thispagestyle`. Na primer,

```
\thispagestyle{empty}
```

uklanja broj i druge elemente samo sa trenutne stranice. Ovakav postupak može biti koristan, na primer, kod naslovne stranice.

Način numerisanja stranica može se promeniti pomoću komande `\pagenumbering`. Za uobičajene arapske brojeve koristi se

```
\pagenumbering{arabic}
```

dok zapis

```
\pagenumbering{roman}
```

koristi male rimske brojeve. Kada je potrebno ručno postaviti broj trenutne stranice, može se promeniti brojač `page`.

```
\setcounter{page}{1}
```

Nakon ove komande trenutna stranica ima broj 1, a naredne stranice nastavljaju numeraciju redom.

Za jednostavno podešavanje zaglavlja i podnožja može se koristiti paket `fancyhdr`. Najpre se paket uključuje u preambuli, a zatim se bira stil `fancy`.

```
\usepackage{fancyhdr}

\pagestyle{fancy}
\fancyhf{}

\fancyhead[L]{Obrazovni softver}
\fancyfoot[C]{\thepage}
```

Komanda `\fancyhf` uklanja prethodno definisani sadržaj zaglavlja i podnožja. Komandom `\fancyhead` određuje se sadržaj zaglavlja. Opcija `[L]` označava njegovu levu stranu, pa se u prikazanom primeru tamo ispisuje tekst `Obrazovni softver`. Slično tome, komanda `\fancyfoot` određuje sadržaj podnožja. Opcija `[C]` označava sredinu, dok komanda `\thepage` predstavlja broj trenutne stranice.

Ova podešavanja mogu se spojiti sa ranije uvedenim podešavanjima dokumenta. Jedan jednostavan primer može izgledati ovako:

```
\documentclass[12pt,a4paper]{article}

\usepackage[margin=2cm,bottom=3cm]{geometry}
\usepackage[T1]{fontenc}
\usepackage[serbian]{babel}
\usepackage{lmodern}
\usepackage{fancyhdr}

\pagestyle{fancy}
\fancyhf{}
\fancyhead[L]{Obrazovni softver}
\fancyfoot[C]{\thepage}
```

```
\begin{document}
```

Ovo je tekst dokumenta.

```
\end{document}
```

U praksi je standardni izgled koji određuje klasa dokumenta često sasvim dovoljan. Kada su potrebna dodatna podešavanja, paketi kao što su **geometry** i **fancyhdr** omogućavaju jednostavno prilagođavanje najvažnijih osobina stranice. Na taj način moguće je podesiti margine, numeraciju, zaglavlje i podnožje, a da se pri tome zadrži osnovna struktura i ujednačen izgled dokumenta. Time smo obuhvatili najvažnije elemente potrebne za pripremu i organizaciju običnog LaTeX dokumenta. U narednoj sekciji prelazimo na mogućnosti LaTeX-a koje su posebno važne pri pisanju matematičkog sadržaja.

2 Matematički sadržaj u LaTeX-u

2.1 Matematički režim

Matematički izrazi u LaTeX-u zapisuju se u posebnom matematičkom režimu. Unutar njega LaTeX drugačije obrađuje slova, brojeve, simbole i razmake kako bi njihov prikaz bio prilagođen matematičkom zapisu.

Kada matematički izraz treba da bude deo običnog reda teksta, može se zapisati između znakova `$`. Na primer,

```
Ako je $x > 0$, onda je $x^2 > 0$.
```

daje rečenicu: Ako je $x > 0$, onda je $x^2 > 0$.

Ovakav način zapisivanja predstavlja matematički režim unutar reda. Početak i kraj matematičkog izraza označeni su znakom `$`, dok se ostatak rečenice obrađuje kao običan tekst.

Isti matematički izraz može se zapisati i pomoću komandi `\(` i `\)`.

```
Ako je \((x > 0)\), onda je \((x^2 > 0)\).
```

Dobijeni rezultat je isti. Oba načina koriste se za matematičke izraze koji se nalaze unutar reda običnog teksta.

Kada matematički izraz treba posebno izdvojiti od okolnog teksta, koristi se izdvojeni matematički režim. Za to se mogu koristiti komande `\[` i `\]`.

```
\[  
x^2 + y^2 = z^2  
\]
```

Dobija se izdvojeni matematički izraz

$$x^2 + y^2 = z^2$$

Ovakav izraz prikazuje se u posebnom redu i odvaja od okolnog teksta. Izrazi zapisani na ovaj način nisu numerisani.

U pojedinim LaTeX dokumentima može se sresti i zapis pomoću dva znaka `$` sa obe strane izraza.

```
$$  
x^2 + y^2 = z^2  
$$
```

I ovaj zapis daje izdvojeni matematički izraz. Ipak, u LaTeX dokumentima preporučuje se korišćenje zapisa `\[... \]`, koji ćemo češće koristiti u nastavku.

Razmaci u matematičkom režimu ne obrađuju se na isti način kao u običnom tekstu. Na primer, zapisi

```
$x+y$  
  
$x \quad + \quad y$
```

daju isti rezultat, $x + y$. LaTeX sam određuje odgovarajuće razmake između matematičkih simbola, pa veći broj razmaka u izvornom fajlu ne menja izgled izraza.

Ponekad je unutar matematičkog izraza potrebno napisati i deo običnog teksta. Ako je uključen paket `amsmath`, za to se može koristiti komanda `\text{...}`.

```
\[
x > 0 \text{ ako i samo ako } -x < 0.
\]
```

Dobija se

$$x > 0 \text{ ako i samo ako } -x < 0.$$

Komanda `\text` omogućava da se navedeni sadržaj unutar matematičkog izraza prikaže kao običan tekst, sa razmacima i odgovarajućim izgledom slova.

Matematički režim predstavlja osnovu za zapisivanje formula u LaTeX-u. Kada je poznato kako se matematički sadržaj umeće u red teksta ili izdvaja u poseban red, mogu se uvoditi različiti simboli i složeniji matematički izrazi. Njih ćemo razmotriti u narednoj podsekciji.

2.2 Matematički simboli i izrazi

Matematički izrazi u LaTeX-u grade se kombinovanjem običnih znakova i posebnih komandi. Mnoge jednostavne operacije zapisuju se neposredno, dok se za složenije simbole i strukture koriste odgovarajuće LaTeX komande.

Indeksi i eksponenti zapisuju se pomoću znakova `_` i `^`. Ako indeks ili eksponent sadrži samo jedan znak, vitičaste zagrade nisu neophodne.

```
\[
x^2 + a_i
\]
```

Dobija se

$$x^2 + a_i.$$

Kada indeks ili eksponent sadrži više znakova, oni se grupišu pomoću vitičastih zagrada.

```
\[
x^{n+1} + a_{i+1}
\]
```

Dobija se

$$x^{n+1} + a_{i+1}.$$

Razlomci se zapisuju pomoću komande `\frac`. Brojilac i imenilac navode se kao njena dva obavezna argumenta.

```
\[
\frac{x+1}{x-1}
\]
```

Dobija se

$$\frac{x+1}{x-1}.$$

Za zapisivanje kvadratnog korena koristi se komanda `\sqrt`.

```
\[
\sqrt{x^2+y^2}
\]
```

Dobija se

$$\sqrt{x^2 + y^2}.$$

Stepen korena može se zadati opcionim argumentom.

```
\[
\sqrt[3]{x}
\]
```

Dobija se

$$\sqrt[3]{x}.$$

Grčka slova zapisuju se komandama koje uglavnom odgovaraju njihovim nazivima. Na primer,

```
\[
\alpha + \beta = \gamma
\]
```

daje

$$\alpha + \beta = \gamma.$$

Neka velika grčka slova zapisuju se komandama čiji naziv počinje velikim slovom.

```
\[
\Delta = b^2 - 4ac
\]
```

Dobija se

$$\Delta = b^2 - 4ac.$$

Pored znakova koji se mogu neposredno napisati sa tastature, LaTeX sadrži komande za veliki broj matematičkih operatora i relacija. Na primer,

```
\[
a \cdot b, \quad a \times b, \quad a \pm b

x \neq y, \quad x \leq y, \quad x \approx z
y \geq z, \quad x \approx z
\]
```

Dobijaju se izrazi

$$a \cdot b, \quad a \times b, \quad a \pm b$$

i

$$x \neq y, \quad x \leq y, \quad y \geq z, \quad x \approx z.$$

Česti su i simboli koji se koriste pri radu sa skupovima.

```
\[
x \in A, \quad y \notin A, \quad A \subseteq B
\]

\[
A \cup B, \quad A \cap B
\]
```

Dobija se

$$x \in A, \quad y \notin A, \quad A \subseteq B$$

i

$$A \cup B, \quad A \cap B.$$

Za zapisivanje logičkih izraza mogu se koristiti odgovarajući logički simboli.

```
\[
\forall x \in A, \quad \exists y \in A
\]

\[
P \land Q \Rightarrow R
\]

\[
P \Leftrightarrow Q, \quad P \lor Q
\]
```

Dobijaju se izrazi

$$\forall x \in A, \quad \exists y \in A$$

$$P \wedge Q \Rightarrow R$$

i

$$P \Leftrightarrow Q, \quad P \vee Q.$$

Standardne matematičke funkcije zapisuju se posebnim komandama. Na primer,

```
\[
\sin x, \quad \cos x, \quad \log x, \quad
\ln x, \quad \exp x
\]
```

Dobija se

$$\sin x, \quad \cos x, \quad \log x, \quad \ln x, \quad \exp x.$$

Korišćenjem ovih komandi nazivi funkcija prikazuju se uspravnim slovima i LaTeX pravilno određuje razmak između funkcije i njenog argumenta. Zbog toga, na primer, funkciju sinus treba zapisati pomoću komande `\sin`, a ne kao tri obična slova `sin` u matematičkom režimu.

Za sume, proizvode, integrale i granične vrednosti koriste se posebni operatori. Donje i gornje granice zapisuju se na isti način kao indeksi i eksponenti.

```
\[
\sum_{k=1}^n k
\]
```

Dobija se

$$\sum_{k=1}^n k.$$

Na sličan način zapisuju se proizvodi.

```
\[
\prod_{k=1}^n k
\]
```

Dobija se

$$\prod_{k=1}^n k.$$

Integral sa granicama može se zapisati ovako:

```
\[
\int_0^1 x^2 dx
\]
```

Dobija se

$$\int_0^1 x^2 dx.$$

Komanda `\,` u ovom primeru pravi mali razmak između integranda i diferencijala.

Za graničnu vrednost koristi se komanda `\lim`.

```
\[
\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1
\]
```

Dobija se

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1.$$

Kod složenijih izraza obične zagrade mogu biti premale u odnosu na sadržaj koji se nalazi između njih. Njihova veličina može se automatski prilagoditi pomoću komandi `\left` i `\right`.

```
\[
\left( \frac{a+b}{c+d} \right)^2
\]
```

Dobija se

$$\left(\frac{a+b}{c+d} \right)^2.$$

Isti postupak može se koristiti sa drugim vrstama zagrada.

```
\[
\left[ \frac{x+1}{x-1} \right]
\]
```

```
\]

\[
\left| \frac{x}{x+1} \right|
\]
```

Dobijaju se izrazi

$$\left[\frac{x+1}{x-1} \right]$$

i

$$\left| \frac{x}{x+1} \right|.$$

Kombinovanjem indeksa, eksponenata, razlomaka, korena, matematičkih simbola i različitih operatora mogu se zapisati veoma složeni izrazi. U narednoj podsekciji razmotrićemo način na koji se jednačine numerišu i kako se složeni izrazi raspoređuju u više redova.

2.3 Jednačine i višeredni izrazi

Izdvojeni matematički izrazi zapisani pomoću komandi `\[ i \]` nisu numerisani. Kada je potrebno da jednačina dobije broj na koji se kasnije možemo pozivati, može se koristiti okruženje `equation`.

```
\begin{equation}
a^2 + b^2 = c^2
\end{equation}
```

Dobija se numerisana jednačina

$$a^2 + b^2 = c^2 \tag{1}$$

Broj jednačine određuje LaTeX, pa ga nije potrebno ručno upisivati. Ako se redosled jednačina kasnije promeni, njihova numeracija automatski se prilagođava.

Složeni matematički izrazi često se prikazuju u više redova. Za njihovo zapisivanje i međusobno poravnanje može se koristiti okruženje `align` iz paketa `amsmath`. Na primer,

```
\begin{align}
(a+b)^2
&= (a+b)(a+b) \\
&= a^2 + ab + ab + b^2 \\
&= a^2 + 2ab + b^2
\end{align}
```

Dobija se

$$(a+b)^2 = (a+b)(a+b) \tag{2}$$

$$= a^2 + ab + ab + b^2 \tag{3}$$

$$= a^2 + 2ab + b^2 \tag{4}$$

Znak `&` određuje mesto po kome se redovi poravnavaju. U ovom primeru postavljen je ispred znaka jednakosti, pa su svi znakovi jednakosti prikazani jedan ispod drugog. Komanda `\\` označava prelazak u novi red unutar okruženja.

Okruženje `align` podrazumevano numeriše svaki njegov red. Kada više redova predstavljaju jednu matematičku celinu koja treba da dobije samo jedan broj, može se koristiti okruženje `split` unutar okruženja `equation`.

```
\begin{equation}
\begin{split}
(a+b)^3
&= (a+b)(a+b)^2 \\
&= (a+b)(a^2+2ab+b^2) \\
&= a^3+3a^2b+3ab^2+b^3
\end{split}
\end{equation}
```

Dobija se

$$\begin{aligned}(a+b)^3 &= (a+b)(a+b)^2 \\ &= (a+b)(a^2+2ab+b^2) \\ &= a^3+3a^2b+3ab^2+b^3\end{aligned}\tag{5}$$

Okruženje `split` omogućava raspoređivanje jednog izraza u više redova, dok okruženje `equation` obezbeđuje jedan zajednički broj jednačine. Znak `&` i ovde određuje mesto poravnanja, a komanda `\\` prelazak u novi red.

Ako nijedan red ne treba da bude numerisan, može se koristiti nenumerisana varijanta `align*`.

```
\begin{align*}
2(x+3)
&= 2x + 6 \\
&= x + x + 6
\end{align*}
```

Dobija se

$$\begin{aligned}2(x+3) &= 2x+6 \\ &= x+x+6\end{aligned}$$

Na numerisanu jednačinu možemo se pozivati iz drugih delova dokumenta. Za to se jednačini najpre dodeljuje oznaka pomoću komande `\label`.

```
\begin{equation}
a^2 + b^2 = c^2
\label{eq:pitagora}
\end{equation}
```

Oznaka `eq:pitagora` služi samo u izvornom fajlu i ne prikazuje se u konačnom dokumentu. Na broj označene jednačine može se pozvati pomoću komande `\eqref`.

```
Iz jednakosti \eqref{eq:pitagora} sledi ...
```

Komanda `\eqref` prikazuje odgovarajući broj zajedno sa zagradama. Ako se broj jednačine promeni, promeniće se i sva pozivanja na nju. Pri korišćenju oznaka i referenci može biti potrebno da se dokument kompajlira više puta kako bi se svi brojevi pravilno povezali.

Okruženja `equation` i `align` omogućavaju numerisanje, poravnanje i organizovanje jednačina koje zauzimaju jedan ili više redova. U narednoj podsekciji razmotrićemo posebne matematičke strukture kao što su matrice i sistemi jednačina.

2.4 Matrice i sistemi jednačina

Matrice se u LaTeX-u mogu zapisivati pomoću odgovarajućih matematičkih okruženja iz paketa `amsmath`. Elementi u istom redu razdvajaju se znakom `&`, dok se novi red započinje komandom `\\`.

Osnovno okruženje `matrix` raspoređuje elemente matrice, ali oko njih ne postavlja zagrade.

```
\[
\begin{matrix}
1 & 2 \\
3 & 4
\end{matrix}
\]
```

Dobija se

$$\begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix}$$

U matematičkom zapisu matrice se najčešće prikazuju unutar odgovarajućih zagrada. Za matricu sa običnim zagradama koristi se okruženje `pmatrix`.

```
\[
A =
\begin{pmatrix}
1 & 2 \\
3 & 4
\end{pmatrix}
\]
```

Dobija se

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}.$$

Ako su potrebne uglaste zagrade, koristi se okruženje `bmatrix`.

```
\[
B =
\begin{bmatrix}
a & b \\
c & d
\end{bmatrix}
\]
```

Dobija se

$$B = \begin{bmatrix} a & b \\ c & d \end{bmatrix}.$$

Za zapis determinante može se koristiti okruženje `vmatrix`, koje elemente postavlja između vertikalnih crta.

```
\[
\begin{vmatrix}
a & b \\
c & d
\end{vmatrix}
= ad-bc
\]
```

Dobija se

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc.$$

Elementi matrice ne moraju biti samo brojevi ili pojedinačne promenljive. Na njihovim mestima mogu se nalaziti i složeniji matematički izrazi.

```
\[
M =
\begin{pmatrix}
x^2 & \frac{1}{x} \\
\sqrt{x} & x+1
\end{pmatrix}
\]
```

Dobija se

$$M = \begin{pmatrix} x^2 & \frac{1}{x} \\ \sqrt{x} & x+1 \end{pmatrix}.$$

Sistemi jednačina mogu se zapisivati pomoću okruženja `cases`. Ono više redova postavlja iza zajedničke leve vitičaste zagrade.

```
\[
\begin{cases}
2x + y = 5, \\
x - y = 1.
\end{cases}
\]
```

Dobija se

$$\begin{cases} 2x + y = 5, \\ x - y = 1. \end{cases}$$

I unutar okruženja `cases` novi red započinje se komandom `\\`. Znak `&` može se koristiti kada delove različitih redova treba međusobno poravnati.

Okruženje `cases` često se koristi i za funkcije koje su definisane različitim izrazima u zavisnosti od vrednosti argumenta.

```
\[
f(x) =
\begin{cases}
x^2, & \text{ako je } x \geq 0, \\
\end{cases}
\]
```

```
-x, & \text{ako je } x < 0.
\end{cases}
\]
```

Dobija se

$$f(x) = \begin{cases} x^2, & \text{ako je } x \geq 0, \\ -x, & \text{ako je } x < 0. \end{cases}$$

U ovom primeru znak `&` razdvaja izraz kojim je funkcija definisana od uslova pod kojim taj izraz važi. Komanda `\text` omogućava da se reči unutar matematičkog zapisa prikazuju kao običan tekst.

Okruženja za matrice i `cases` omogućavaju pregledno zapisivanje matematičkih struktura koje sadrže više redova i kolona. U narednoj podsekciji razmotrićemo različite načine prikazivanja matematičkih oznaka pomoću odgovarajućih fontova.

2.5 Matematički fontovi

Slova koja se zapisuju neposredno u matematičkom režimu LaTeX podrazumevano prikazuje matematičkim kurzivom. Takav zapis najčešće se koristi za promenljive.

```
\[
x + y = a + b
\]
```

Dobija se

$$x + y = a + b.$$

Kada slova unutar matematičkog izraza treba prikazati uspravno, može se koristiti komanda `\mathrm`.

```
\[
x \quad \mathrm{x}
\]
```

Dobijaju se dva različita zapisa

$$x \quad \mathbf{x}.$$

Prvo slovo predstavlja uobičajenu matematičku promenljivu i zato je prikazano kurzivom, dok komanda `\mathrm` daje uspravna slova. Ovakav zapis može se koristiti kada oznaka u matematičkom izrazu ne predstavlja običnu promenljivu.

Za podebljana latinična slova koristi se komanda `\mathbf`. Takav zapis se, između ostalog, može koristiti za označavanje vektora.

```
\[
\mathbf{v} = (v_1, v_2, v_3)
\]
```

Dobija se

$$\mathbf{v} = (v_1, v_2, v_3).$$

Kaligrafska velika slova mogu se zapisati pomoću komande `\mathcal`. Ona se često koriste za označavanje familija skupova, prostora i drugih matematičkih objekata.

```
\[
\mathcal{A},\quad \mathcal{F},\quad \mathcal{P}
\]
```

Dobija se

$$\mathcal{A}, \quad \mathcal{F}, \quad \mathcal{P}.$$

Komanda `\mathcal` koristi se za velika latinična slova.

Za neke standardne skupove brojeva uobičajeno se koriste dvostruko iscrtana slova. Njih omogućava komanda `\mathbb`, dostupna kada je uključen paket `amssymb`.

```
\[
\mathbb{N},\quad
\mathbb{Z},\quad
\mathbb{Q},\quad
\mathbb{R},\quad
\mathbb{C}
\]
```

Dobijaju se standardne oznake

$$\mathbb{N}, \quad \mathbb{Z}, \quad \mathbb{Q}, \quad \mathbb{R}, \quad \mathbb{C}.$$

Izbor matematičkog fonta često ima i značenje, a ne samo uticaj na izgled zapisa. Promenljive se uobičajeno prikazuju kurzivom, podebljana slova mogu označavati vektore, kaligrafska slova posebne familije objekata, dok se dvostruko iscrtana slova često koriste za standardne skupove brojeva. U narednoj podsekciji razmotrićemo način na koji se u LaTeX dokumentima organizuju definicije, teoreme i njihovi dokazi.

2.6 Definicije, teoreme i dokazi

Matematički dokumenti često sadrže definicije, teoreme, leme, posledice i njihove dokaze. Umesto da se naziv, numeracija i izgled svakog takvog elementa podešavaju ručno, LaTeX omogućava da se za njih definišu posebna okruženja. Za tu namenu može se koristiti paket `amsthm`.

Paket se uključuje u preambuli dokumenta.

```
\usepackage{amsthm}
```

Nova okruženja za matematičke iskaze definišu se pomoću komande `\newtheorem`. Njihov izgled može se odrediti komandom `\theoremstyle`. Za definicije je pogodan stil `definition`.

```
\theoremstyle{definition}
\newtheorem{definicija}{Definicija}[section]
```

Prvi argument komande `\newtheorem` određuje naziv okruženja koji se koristi u izvornom fajlu, dok drugi određuje naziv koji će biti prikazan u dokumentu. Opcija `[section]` određuje da numeracija zavisi od broja sekcije.

Nakon ovakve definicije okruženje se može koristiti na sledeći način.

```
\begin{definicija}
Ceo broj  $n$  je paran ako postoji  $k \in \mathbb{Z}$ 
```

```
takav da je  $n=2k$ .  
\end{definicija}
```

Dobija se

Definicija 2.1. Ceo broj n je paran ako postoji $k \in \mathbb{Z}$ takav da je $n = 2k$.

LaTeX automatski ispisuje naziv definicije i njen broj. Ako se pre nje dodaju ili uklone drugi numerisani iskazi, numeracija se automatski prilagođava.

Za teoreme se najčešće koristi stil `plain`. Teoremu možemo povezati sa brojačem koji već koristi okruženje `definicija`.

```
\theoremstyle{plain}  
\newtheorem{teorema}[definicija]{Teorema}
```

Opcija `[definicija]` određuje da definicije i teoreme koriste istu numeraciju. Ako je prethodni iskaz imao broj 2.1, sledeća teorema imaće broj 2.2.

Teorema se zatim zapisuje pomoću odgovarajućeg okruženja.

```
\begin{teorema}  
Zbir dva parna cela broja je paran.  
\end{teorema}
```

Dobija se

Teorema 2.2. *Zbir dva parna cela broja je paran.*

Stil `plain` podrazumevano ističe tekst matematičkog iskaza kurzivom, dok stil `definition` koristi uspravan tekst. Na taj način različite vrste matematičkog sadržaja mogu imati ujednačen i prepoznatljiv izgled u celom dokumentu.

Paket `amsthm` obezbeđuje i okruženje `proof` namenjeno pisanju dokaza. Njegov podrazumevani naziv može se promeniti u preambuli.

```
\renewcommand{\proofname}{Dokaz}
```

Dokaz prethodne teoreme može se zapisati ovako.

```
\begin{proof}  
Neka su  $a$  i  $b$  parni celi brojevi. Tada postoje  
 $p, q \in \mathbb{Z}$  takvi da je  $a=2p$  i  $b=2q$ .  
Zato je  

$$a+b=2p+2q=2(p+q).$$
  
Pošto je  $p+q \in \mathbb{Z}$ , broj  $a+b$  je paran.  
\end{proof}
```

Dobija se

Dokaz. Neka su a i b parni celi brojevi. Tada postoje $p, q \in \mathbb{Z}$ takvi da je $a = 2p$ i $b = 2q$. Zato je

$$a + b = 2p + 2q = 2(p + q).$$

Pošto je $p + q \in \mathbb{Z}$, broj $a + b$ je paran. □

Okruženje `proof` automatski prikazuje naziv dokaza i oznaku njegovog završetka. Zbog toga ove elemente nije potrebno ručno dodavati.

Na isti način mogu se definisati i druga okruženja, kao što su leme i posledice. Na primer,

```
\newtheorem{lema}[definicija]{Lema}  
\newtheorem{posledica}[definicija]{Posledica}
```

Ova okruženja koriste isti brojač kao definicije i teoreme, pa svi odgovarajući matematički iskazi mogu imati zajedničku numeraciju.

Posebna okruženja za definicije, teoreme i dokaze omogućavaju da matematički tekst bude jasno organizovan i dosledno oblikovan. Zajedno sa načinima zapisivanja formula, jednačina, matrica i drugih matematičkih izraza, ona predstavljaju osnovu za pripremu strukturiranih matematičkih dokumenata u LaTeX-u.

3 Dodatni elementi LaTeX dokumenta

3.1 Slike

LaTeX omogućava uključivanje postojećih grafičkih fajlova u dokument. Za rad sa slikama najčešće se koristi paket `graphicx`, koji se uključuje u preambuli.

```
\usepackage{graphicx}
```

Nakon uključivanja paketa slika se može umetnuti pomoću komande `\includegraphics`. Ako se fajl slike nalazi u istom direktorijumu kao i LaTeX dokument, dovoljno je navesti njegovo ime.

```
\includegraphics{slika.png}
```

Komanda umeće sadržaj fajla `slika.png` na mesto na kome je navedena. Ako se slika nalazi u drugom direktorijumu, može se navesti i odgovarajuća relativna putanja.

```
\includegraphics{slike/grafik.png}
```

Veličina slike može se podesiti opcionim argumentima komande `\includegraphics`. Često je korisno zadati širinu slike u odnosu na širinu teksta.

```
\includegraphics[width=0.6\textwidth]{slika.png}
```

Vrednost `0.6\textwidth` određuje da širina slike bude jednaka 60 procenata širine prostora namenjenog tekstu. Visina se pri tome automatski prilagođava tako da se sačuvaju proporcije slike.

Kada slika treba da predstavlja poseban numerisani element dokumenta, obično se postavlja unutar okruženja `figure`. Ovo okruženje omogućava centriranje slike, dodavanje naslova i automatsku numeraciju.

```
\begin{figure}[h]
\centering
\includegraphics[width=0.6\textwidth]{slika.png}
\caption{Primer slike}
\end{figure}
```

Komanda `\centering` centrira sadržaj okruženja, dok komanda `\caption` dodaje naslov slike. LaTeX automatski dodaje i broj slike, pa numeraciju nije potrebno ručno upisivati.

Okruženje `figure` spada u plutajuća okruženja. To znači da LaTeX može promeniti položaj slike kako bi bolje rasporedio sadržaj stranice. Opcija `[h]` označava da sliku želimo približno na mestu na kome se okruženje nalazi u izvornom fajlu. Mogu se koristiti i druge oznake, kao što su `t` za vrh stranice, `b` za dno stranice i `p` za posebnu stranicu namenjenu plutajućim elementima.

Slici se može dodeliti i oznaka pomoću komande `\label`. Ona se postavlja nakon komande `\caption`.

```
\begin{figure}[h]
\centering
\includegraphics[width=0.6\textwidth]{slika.png}
\caption{Primer slike}
\label{fig:primer}
```

```
\end{figure}
```

Oznaka `fig:primer` omogućava da se iz drugih delova dokumenta pozovemo na broj ove slike. Način korišćenja oznaka i referenci detaljnije ćemo razmotriti u posebnoj podsekciji.

Komanda `\includegraphics` služi za samo umetanje grafičkog fajla, dok okruženje `figure` omogućava da slika postane numerisani element dokumenta sa naslovom i oznakom. U narednoj podsekciji razmotrićemo sličan način organizovanja tabelarnih podataka.

3.2 Tabele

Tabelarni podaci u LaTeX-u mogu se organizovati pomoću okruženja `tabular`. Pri njegovom započinjanju navodi se broj kolona i način poravnanja njihovog sadržaja. Oznaka `l` predstavlja poravnanje ulevo, `c` centriranje, dok `r` predstavlja poravnanje udesno.

Jednostavna tabela sa tri kolone može se zapisati ovako:

```
\begin{tabular}{lcr}
Ime & Poeni & Ocena \\
Ana & 85 & 9 \\
Marko & 72 & 8
\end{tabular}
```

Dobija se:

Ime	Poeni	Ocena
Ana	85	9
Marko	72	8

Znak `&` razdvaja sadržaj susednih kolona, dok komanda `\\` označava kraj jednog reda i prelazak na sledeći. U ovom primeru prva kolona poravnata je ulevo, druga je centrirana, a treća udesno.

Po potrebi se u tabelu mogu dodati horizontalne i vertikalne linije. Vertikalne linije zadaju se znakom `|` u opisu kolona, dok se horizontalna linija dodaje komandom `\hline`.

```
\begin{tabular}{|l|c|c|}
\hline
Ime & Poeni & Ocena \\
\hline
Ana & 85 & 9 \\
Marko & 72 & 8 \\
\hline
\end{tabular}
```

Dobija se:

Ime	Poeni	Ocena
Ana	85	9
Marko	72	8

Okruženje `tabular` određuje raspored redova i kolona, ali samo po sebi ne daje tabeli naslov ni broj. Kada tabela treba da predstavlja poseban numerisani element dokumenta, okruženje `tabular` može se postaviti unutar okruženja `table`.

```

\begin{table}[h]
\centering
\begin{tabular}{lcc}
Ime & Poeni & Ocena \\
Ana & 85 & 9 \\
Marko & 72 & 8
\end{tabular}
\caption{Rezultati studenata}
\end{table}

```

Dobija se

Ime	Poeni	Ocena
Ana	85	9
Marko	72	8

Tabela 1: Rezultati studenata

Komanda `\centering` centrirala tabelu, dok komanda `\caption` dodaje njen naslov. LaTeX automatski određuje broj tabele, pa numeraciju nije potrebno ručno upisivati. Kao i okruženje `figure`, okruženje `table` je plutajuće, pa LaTeX po potrebi može prilagoditi njegov položaj.

Tabeli se može dodeliti i oznaka pomoću komande `\label`. Ona se postavlja nakon komande `\caption`.

```

\begin{table}[h]
\centering
\begin{tabular}{lcc}
Ime & Poeni & Ocena \\
Ana & 85 & 9 \\
Marko & 72 & 8
\end{tabular}
\caption{Rezultati studenata}
\label{tab:rezultati}
\end{table}

```

Oznaka `tab:rezultati` omogućava da se iz drugih delova dokumenta pozovemo na broj tabele. Način korišćenja ovakvih oznaka razmotrićemo detaljnije u narednoj podsekciji.

Okruženje `tabular` služi za organizovanje podataka u redove i kolone, dok okruženje `table` omogućava da takav sadržaj postane numerisani element dokumenta sa naslovom i oznakom. U narednoj podsekciji detaljnije ćemo razmotriti oznake i reference koje omogućavaju povezivanje različitih delova dokumenta.

3.3 Oznake i reference

LaTeX automatski numerise različite elemente dokumenta, kao što su sekcije, jednačine, slike i tabele. Kada se na neki od tih elemenata pozivamo iz drugog dela teksta, njegov broj nije potrebno ručno upisivati. Umesto toga, elementu se može dodeliti oznaka, a LaTeX zatim automatski umeće odgovarajući broj.

Oznaka se dodeljuje pomoću komande `\label`, čiji argument predstavlja interno ime označenog elementa. Na primer, jednačina se može označiti na sledeći način.

```
\begin{equation}
a^2 + b^2 = c^2
\label{eq:pitagora}
\end{equation}
```

Komanda `\label{eq:pitagora}` ne prikazuje dodatni tekst u dokumentu. Oznaka `eq:pitagora` služi samo za povezivanje ovog elementa sa drugim delovima izvornog fajla.

Na broj označenog elementa može se pozvati pomoću komande `\ref`.

Kao što se vidi iz jednačine `\ref{eq:pitagora}`, ...

LaTeX umesto komande `\ref` prikazuje odgovarajući broj jednačine. Za matematičke jednačine često je pogodnije koristiti komandu `\eqref` iz paketa `amsmath`, koja broj automatski prikazuje u zagradama.

Iz jednačine `\eqref{eq:pitagora}` sledi ...

Isti princip može se koristiti za različite numerisane elemente dokumenta. Sekcija se može označiti tako što se komanda `\label` navede neposredno nakon njenog naslova.

```
\section{Teorija grafova}
\label{sec:grafovi}
```

Na njen broj se kasnije možemo pozvati pomoću komande

Kao što je opisano u sekciji `\ref{sec:grafovi}`, ...

Kod slika i tabela oznaka se obično postavlja nakon komande `\caption`, kako bi bila povezana sa odgovarajućim brojem.

```
\begin{figure}[h]
\centering
\includegraphics[width=0.6\textwidth]{slika.png}
\caption{Primer slike}
\label{fig:primer}
\end{figure}
```

Na sliku se zatim možemo pozvati zapisom

Na slici `\ref{fig:primer}` prikazan je ...

Na isti način može se označiti i tabela.

```
\begin{table}[h]
\centering
\begin{tabular}{lc}
Ime & Poeni \\
Ana & 85 \\
Marko & 72
\end{tabular}
\caption{Rezultati studenata}
\label{tab:rezultati}
```

```
\end{table}
```

Na njen broj se možemo pozvati pomoću komande `\ref`.

```
Rezultati su prikazani u tabeli  
\ref{tab:rezultati}.
```

Pored broja označenog elementa, može se dobiti i broj stranice na kojoj se on nalazi. Za to se koristi komanda `\pageref`.

```
Tabela se nalazi na strani  
\pageref{tab:rezultati}.
```

LaTeX umesto komande `\pageref` umeće broj odgovarajuće stranice.

Nazivi oznaka mogu se birati proizvoljno, ali je korisno koristiti prefikse koji ukazuju na vrstu označenog elementa. Na primer, za sekcije se može koristiti `sec:`, za slike `fig:`, za tabele `tab:`, a za jednačine `eq:`. Tako oznake kao što su `sec:grafovi`, `fig:primer`, `tab:rezultati` i `eq:pitagora` jasno ukazuju na element na koji se odnose.

Pri korišćenju oznaka i referenci može biti potrebno da se dokument kompajlira više puta. Pri prvom kompajliranju LaTeX određuje brojeve i položaje označenih elemenata, a pri narednom ih koristi za pravilno prikazivanje referenci.

Glavna prednost ovakvog načina povezivanja elemenata jeste to što njihove brojeve nije potrebno ručno održavati. Ako se redosled sekcija, slika, tabela ili jednačina promeni, LaTeX automatski prilagođava njihove brojeve i sve reference koje se na njih odnose. U narednoj podsekciji razmotrićemo hiperveze koje omogućavaju dodatno povezivanje delova dokumenta i spoljnih sadržaja.

3.4 Hiperveze

LaTeX dokument može sadržati hiperveze koje omogućavaju prelazak na druge delove istog dokumenta ili na sadržaje dostupne na vebu. Za rad sa hipervezama najčešće se koristi paket `hyperref`.

Paket se može uključiti u preambuli na sledeći način:

```
\usepackage[unicode,hidelinks]{hyperref}
```

Opcija `unicode` omogućava pravilnu obradu Unicode znakova u elementima koje paket koristi, dok `hidelinks` uklanja upadljive okvire i boje oko veza. Same veze pri tome ostaju aktivne.

Paket `hyperref` povezuje se i sa oznakama i referencama koje smo prethodno razmatrali. Kada je paket uključen, reference napravljene pomoću komandi kao što su `\ref`, `\eqref` i `\pageref` postaju aktivne veze u PDF dokumentu. Klikom na njih može se preći na odgovarajući deo dokumenta.

Paket omogućava i jednostavno navođenje veb-adresa. Ako želimo da se u dokumentu prikaže sama adresa, koristi se komanda `\url`.

```
\url{https://www.matf.bg.ac.rs}
```

Dobija se `https://www.matf.bg.ac.rs`. Prikazana adresa je istovremeno i aktivna veza koja vodi ka navedenoj veb-stranici.

Kada ne želimo da se prikazuje sama adresa, već tekst koji opisuje vezu, može se koristiti komanda `\href`. Prvi argument predstavlja adresu, a drugi tekst koji će biti prikazan u

dokumentu.

```
\href{https://www.matf.bg.ac.rs}{Matematički fakultet}
```

Dobija se tekst Matematički fakultet, koji predstavlja aktivnu hipervezu ka veb-stranici Matematičkog fakulteta.

Na ovaj način dugačke ili manje pregledne adrese mogu se zameniti tekстом koji jasno opisuje njihov sadržaj. Sama adresa i dalje ostaje sačuvana kao odredište hiperveze.

Paket `hyperref` obično se uključuje pri kraju liste paketa u preambuli, jer njegove mogućnosti utiču na različite elemente dokumenta, uključujući sadržaj, reference i spoljne veze.

Hiperveze omogućavaju jednostavno povezivanje delova dokumenta i pristup spoljnim sadržajima, dok se postojeće oznake i reference mogu automatski pretvoriti u aktivne veze. U narednoj podsekciji razmotrićemo navođenje literature i povezivanje izvora sa odgovarajućim mestima u tekstu.

3.5 Literatura i citiranje

Naučni i stručni dokumenti često se zasnivaju na različitim knjigama, člancima, veb-stranicama i drugim izvorima. Korišćeni izvori mogu se navesti u posebnom spisku literature, dok se iz odgovarajućih delova teksta na njih upućuje pomoću citata.

Jednostavan način za navođenje literature u LaTeX-u jeste okruženje `thebibliography`. Svaki izvor unutar njega uvodi se komandom `\bibitem`.

```
\begin{thebibliography}{9}

\bibitem{lamport}
Leslie Lamport,
\textit{LaTeX: A Document Preparation System},
Addison-Wesley, 1994.

\bibitem{overleaf}
Overleaf,
\textit{Learn LaTeX in 30 minutes},
\url{https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes}.

\end{thebibliography}
```

Svaka komanda `\bibitem` sadrži ključ koji predstavlja internu oznaku izvora. U ovom primeru koriste se ključevi `lamport` i `overleaf`. Oni se koriste u izvornom fajlu, dok LaTeX u konačnom dokumentu automatski prikazuje odgovarajuće oznake stavki literature.

Argument `{9}` okruženja `thebibliography` određuje širinu prostora predviđenog za oznake izvora. Za spisak sa manje od deset numerisanih stavki ovakav zapis je dovoljan.

Na određeni izvor poziva se pomoću komande `\cite`. Na primer,

```
Osnovne mogućnosti LaTeX-a detaljno su opisane
u \cite{lamport}.
```

LaTeX umesto komande `\cite` prikazuje oznaku odgovarajuće stavke iz spiska literature. Na taj način citat ostaje povezan sa izvorom čak i ako se redosled stavki u literaturi kasnije promeni.

Jednom komandom moguće je citirati i više izvora.

Dodatni primeri mogu se pronaći u izvorima
`\cite{lamport,overleaf}`.

LaTeX tada prikazuje oznake svih navedenih izvora u okviru jednog citata.

Okruženje `thebibliography` obično se postavlja pri kraju dokumenta. Ako je uključen paket `hyperref`, citati u PDF dokumentu postaju aktivne veze ka odgovarajućim stavkama literature, dok se pomoću komande `\url` mogu navoditi aktivne veze ka izvorima dostupnim na vebu.

Kod većih dokumenata sa velikim brojem izvora mogu se koristiti i posebni sistemi za automatizovano upravljanje bibliografijom. Za kraće dokumente okruženje `thebibliography`, zajedno sa komandama `\bibitem` i `\cite`, pruža dovoljno mogućnosti za organizovanje literature i povezivanje izvora sa odgovarajućim delovima teksta.

Slike, tabele, reference, hiperveze i literatura dopunjuju osnovni tekst i omogućavaju izradu preglednih i dobro povezanih LaTeX dokumenata. U narednoj sekciji upoznaćemo TikZ, koji omogućava da se grafički prikazi konstruišu neposredno u LaTeX dokumentu.

4 Grafički prikazi u TikZ-u

4.1 Osnovna struktura crteža

TikZ je sistem za izradu grafičkih prikaza neposredno u LaTeX dokumentu. Za razliku od umetanja postojeće slike iz posebnog grafičkog fajla, kod TikZ-a se elementi crteža opisuju odgovarajućim komandama, a konačan prikaz generiše se zajedno sa ostatkom dokumenta.

Za korišćenje TikZ-a potrebno je u preambuli uključiti paket `tikz`.

```
\usepackage{tikz}
```

Jedan TikZ crtež zapisuje se unutar okruženja `tikzpicture`. Najjednostavniji primer može sadržati samo jednu duž.

```
\begin{tikzpicture}
\draw (0,0) -- (3,0);
\end{tikzpicture}
```

Dobija se

Komanda `\draw` koristi se za crtanje putanje. U ovom primeru tačke $(0,0)$ i $(3,0)$ predstavljaju krajeve duži, dok zapis `--` određuje da se između njih nacrtava prava linija. Naredba `\draw` završava se znakom `;`.

Koordinate se zapisuju u obliku (x,y) . Prva vrednost određuje horizontalni, a druga vertikalni položaj tačke. Ako nisu uvedena dodatna podešavanja, jedna jedinica u ovom koordinatnom sistemu odgovara jednom centimetru.

Unutar jednog okruženja `tikzpicture` može se nalaziti više naredbi za crtanje. Na primer,

```
\begin{tikzpicture}
\draw (0,0) -- (3,0);
\draw (0,0) -- (0,2);
\end{tikzpicture}
```

Dobija se



Obe naredbe pripadaju istom crtežu i koriste isti koordinatni sistem. Na ovaj način složeniji grafički prikazi mogu se postepeno graditi dodavanjem novih elemenata unutar istog okruženja.

Okruženje `tikzpicture`, koordinate i komanda `\draw` predstavljaju osnovu za izradu TikZ crteža. U narednoj podsekciji detaljnije ćemo razmotriti zadavanje tačaka i različite načine povezivanja koordinata linijama.

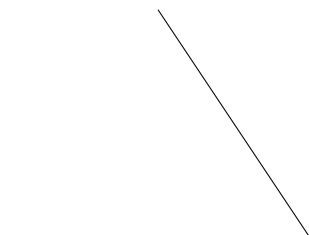
4.2 Koordinate, tačke i linije

Osnovni elementi TikZ crteža mogu se zadavati pomoću tačaka određenih koordinatama i linija koje ih povezuju. U jednoj naredbi `\draw` može se navesti više tačaka, pa se na taj način mogu jednostavno konstruisati izlomljene linije i mnogouglovi.

Na primer, tri tačke mogu se povezati na sledeći način:

```
\begin{tikzpicture}
\draw (0,0) -- (4,0) -- (2,3);
\end{tikzpicture}
```

Dobija se

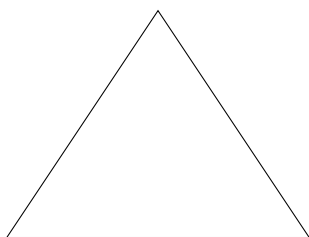


Zapis `--` između dve koordinate određuje da se odgovarajuće tačke povežu pomoću duži. Kada se na isti način navede više koordinata, svaka nova tačka povezuje se sa prethodnom.

Ako poslednju tačku želimo da povežemo sa početnom i tako zatvorimo putanju, može se koristiti ključna reč `cycle`. Na primer, trougao se može nacrtati ovako.

```
\begin{tikzpicture}
\draw (0,0) -- (4,0) -- (2,3) -- cycle;
\end{tikzpicture}
```

Dobija se

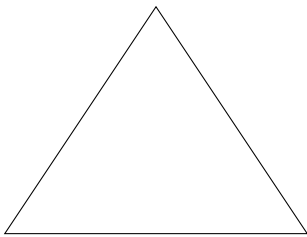


Kod složenijih crteža često je preglednije da se važnim tačkama najpre dodele imena. Za to se koristi komanda `\coordinate`.

```
\begin{tikzpicture}
\coordinate (A) at (0,0);
\coordinate (B) at (4,0);
\coordinate (C) at (2,3);

\draw (A) -- (B) -- (C) -- cycle;
\end{tikzpicture}
```

Dobija se isti trougao



Komanda `\coordinate` ne crta samu tačku, već samo dodeljuje ime određenoj koordinati. Nakon toga se umesto numeričkih koordinata mogu koristiti imena kao što su (A) , (B) i (C) . Ovakav zapis je naročito koristan kada se ista tačka koristi u više delova crteža.

Ako želimo da tačka bude i vidljivo označena na crtežu, može se koristiti komanda `\fill`. Na primer,

```
\begin{tikzpicture}
\fill (0,0) circle (2pt);
\fill (3,1) circle (2pt);
\end{tikzpicture}
```

Dobija se



U ovom primeru komanda `\fill` ispunjava kružić poluprečnika `2pt` i na taj način prikazuje zadatu tačku. Tekstualne oznake tačaka, kao što su A , B i C , razmotrićemo kasnije.

Linije mogu imati i strelice. Njihov smer zadaje se opcionim argumentom komande `\draw`.

```
\begin{tikzpicture}
\draw[->] (0,0) -- (4,0);
\end{tikzpicture}
```

Dobija se

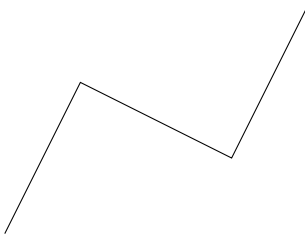


Opcija `[->]` postavlja strelicu na kraj linije. Na sličan način može se koristiti `[<-]` za strelicu na početku ili `[<->]` za strelice na oba kraja.

Više duži može se povezati i u jednu izlomljenu liniju.

```
\begin{tikzpicture}
\draw (0,0) -- (1,2) -- (3,1) -- (4,3);
\end{tikzpicture}
```

Dobija se



Koordinate, imenovane tačke i linije predstavljaju osnovne gradivne elemente TikZ crteža. Njihovim povezivanjem mogu se konstruisati različite geometrijske figure i putanje. U narednoj podsekciji razmotrićemo komande za neposredno crtanje pravougaonika, kružnica, lukova i drugih osnovnih geometrijskih oblika.

4.3 Geometrijski oblici i putanje

Pored povezivanja pojedinačnih tačaka linijama, TikZ sadrži i posebne konstrukcije za crtanje čestih geometrijskih oblika. One se koriste kao delovi putanje koju iscrtava komanda `\draw`.

Pravougaonik se može nacrtati pomoću ključne reči **rectangle**. Navode se koordinate dva njegova naspramna temena.

```
\begin{tikzpicture}
\draw (0,0) rectangle (4,2);
\end{tikzpicture}
```

Dobija se

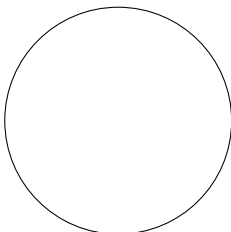


U ovom primeru tačke $(0,0)$ i $(4,2)$ određuju dva naspramna temena pravougaonika.

Kružnica se crta pomoću ključne reči **circle**. Najpre se navodi koordinata njenog centra, a zatim poluprečnik.

```
\begin{tikzpicture}
\draw (0,0) circle (1.5);
\end{tikzpicture}
```

Dobija se

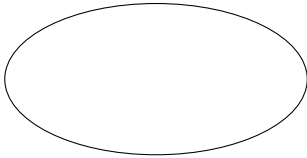


Centar kružnice nalazi se u koordinatnom početku, dok njen poluprečnik iznosi 1.5 jedinica.

Na sličan način može se nacrtati i elipsa pomoću ključne reči `ellipse`. Zadaju se njena horizontalna i vertikalna poluosa.

```
\begin{tikzpicture}
\draw (0,0) ellipse (2 and 1);
\end{tikzpicture}
```

Dobija se

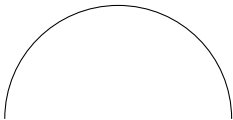


U ovom primeru horizontalna poluosa ima dužinu 2, a vertikalna poluosa dužinu 1.

TikZ omogućava i crtanje delova kružnice. Za to se koristi konstrukcija `arc`, kojoj se mogu zadati početni ugao, završni ugao i poluprečnik.

```
\begin{tikzpicture}
\draw (0,0)
  arc[start angle=180,end angle=0,radius=1.5];
\end{tikzpicture}
```

Dobija se



U ovom primeru luk počinje u zadatoj početnoj tački, a vrednosti `start angle` i `end angle` određuju uglove između kojih se on iscrtava. Opcija `radius` određuje poluprečnik odgovarajuće kružnice.

Niz tačaka, linija i geometrijskih konstrukcija koje komanda `\draw` povezuje predstavlja putanju. Putanja može biti otvorena ili zatvorena, kao kod mnogouglova koje smo ranije zatvarali pomoću ključne reči `cycle`.

Zatvorena putanja može se i popuniti. Za to se umesto komande `\draw` može koristiti komanda `\fill`.

```
\begin{tikzpicture}
\fill (0,0) -- (3,0) -- (1.5,2) -- cycle;
\end{tikzpicture}
```

Dobija se



Komanda `\fill` popunjava unutrašnjost zatvorene putanje. U ovom primeru tri tačke odre-

đuju trougao, dok ključna reč `cycle` povezuje poslednju tačku sa početnom i zatvara njegovu unutrašnjost.

Pravougaonici, kružnice, elipse, lukovi i proizvoljne putanje omogućavaju izgradnju različitih geometrijskih prikaza. Da bi takav crtež imao jasno matematičko značenje, njegovim tačkama i drugim elementima često je potrebno dodati odgovarajuće oznake. Njih ćemo razmotriti u narednoj podsekciji.

4.4 Tekst i matematičke oznake

Geometrijski crtež često treba dopuniti nazivima tačaka, oznakama dužina, matematičkim izrazima i drugim tekstualnim sadržajem. Za postavljanje takvih oznaka u TikZ-u koristi se komanda `\node`.

Najjednostavniji način je da se navede koordinata na kojoj tekst treba da bude postavljen.

```
\begin{tikzpicture}
\node at (0,0) {Centar};
\end{tikzpicture}
```

Dobija se

Centar

Tekst koji treba prikazati navodi se između vitičastih zagrada, dok konstrukcija `at (0,0)` određuje njegov položaj na crtežu.

Položaj teksta može se dodatno odrediti opcionim argumentima kao što su `above`, `below`, `left` i `right`. Na primer,

```
\begin{tikzpicture}
\fill (0,0) circle (2pt);
\node[above] at (0,0) {Tačka};
\end{tikzpicture}
```

Dobija se

Tačka
•

Opcija `above` postavlja tekst iznad zadate koordinate. Na isti način mogu se koristiti opcije `below`, `left` i `right`, kao i njihove kombinacije poput `above right`.

Komanda `\node` može se povezati neposredno sa elementom koji se crta. Na primer, tačku možemo nacrtati i označiti jednom naredbom.

```
\begin{tikzpicture}
\fill (0,0) circle (2pt) node[below left] {$A$};
\end{tikzpicture}
```

Dobija se

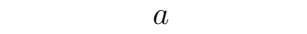
A
•

U ovom primeru čvor je vezan za nacrtanu tačku, pa nije potrebno ponovo navoditi njenu koordinatu. Oznaka A zapisana je u matematičkom režimu, što je uobičajeno za nazive tačaka i druge matematičke oznake.

Čvorovi se mogu postavljati i duž nacrtane linije. Opcija `midway` postavlja čvor približno na sredinu odgovarajućeg dela putanje.

```
\begin{tikzpicture}
\draw (0,0) -- (4,0) node[midway,above] {$a$};
\end{tikzpicture}
```

Dobija se



Opcija `midway` određuje položaj čvora duž linije, dok `above` postavlja oznaku iznad nje. Na ovaj način mogu se jednostavno dodavati oznake dužina i drugih geometrijskih elemenata.

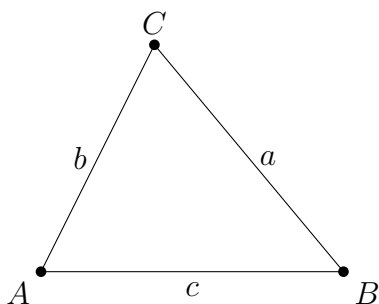
Kombinovanjem imenovanih koordinata, tačaka i čvorova može se napraviti potpuno označen geometrijski crtež. Na primer,

```
\begin{tikzpicture}
\coordinate (A) at (0,0);
\coordinate (B) at (4,0);
\coordinate (C) at (1.5,3);

\draw (A) -- (B) node[midway,below] {$c$}
      -- (C) node[midway,right] {$a$}
      -- (A) node[midway,left] {$b$};

\fill (A) circle (2pt) node[below left] {$A$};
\fill (B) circle (2pt) node[below right] {$B$};
\fill (C) circle (2pt) node[above] {$C$};
\end{tikzpicture}
```

Dobija se



U ovom primeru temena trougla predstavljena su imenovanim koordinatama, a njihove oznake dodate su pomoću čvorova. Čvorovi postavljeni duž stranica koriste se za oznake a , b i c . Na taj način tekstualni i matematički sadržaj postaje sastavni deo samog crteža.

Komanda `\node` omogućava da se geometrijskim elementima dodaju nazivi, matematičke oznake i druga objašnjenja. U narednoj podsekciji razmotrićemo načine za dodatno oblikovanje crteža pomoću različitih stilova, debljina linija i boja.

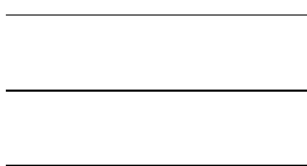
4.5 Stilovi i oblikovanje crteža

Izgled elemenata TikZ crteža može se menjati pomoću opcija koje se navode između uglastih zagrada. Na taj način mogu se podesiti debljina i vrsta linije, boja, način ispunjavanja zatvorenih oblasti i druge osobine crteža.

Debljina linije može se menjati pomoću opcija kao što su `thin`, `thick` i `very thick`.

```
\begin{tikzpicture}
\draw[thin] (0,2) -- (4,2);
\draw[thick] (0,1) -- (4,1);
\draw[very thick] (0,0) -- (4,0);
\end{tikzpicture}
```

Dobija se

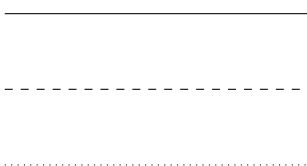


Različite debljine mogu se koristiti za isticanje važnijih elemenata crteža ili razlikovanje glavnih i pomoćnih linija.

Pored pune linije, TikZ omogućava i druge načine njenog prikazivanja. Među najčešćim su isprekidana linija `dashed` i tačkasta linija `dotted`.

```
\begin{tikzpicture}
\draw (0,2) -- (4,2);
\draw[dashed] (0,1) -- (4,1);
\draw[dotted] (0,0) -- (4,0);
\end{tikzpicture}
```

Dobija se



Isprekidane i tačkaste linije naročito su korisne za pomoćne elemente geometrijskog crteža, jer se na taj način mogu razlikovati od njegovih osnovnih delova.

Boja linije može se zadati opcijom `draw`. Na primer,

```
\begin{tikzpicture}
\draw[draw=blue,thick] (0,0) -- (4,0);
\end{tikzpicture}
```

Dobija se



Više opcija može se navesti istovremeno, pri čemu se one razdvajaju zarezima. U prethodnom primeru linija je istovremeno obojena i podebljana.

Kod zatvorenih oblika može se odrediti i boja njihove unutrašnjosti. Za to se koristi opcija `fill`.

```
\begin{tikzpicture}
\fill[fill=gray!20]
  (0,0) -- (4,0) -- (2,3) -- cycle;
\end{tikzpicture}
```

Dobija se

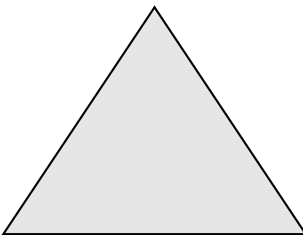


Zapis `gray!20` predstavlja svetlu nijansu sive boje. Komanda `\fill` ispunjava unutrašnjost zatvorene putanje, ali ne mora posebno da istakne njenu granicu.

Kada želimo istovremeno da ispunimo oblast i nacrtamo njenu granicu, može se koristiti komanda `\filldraw`.

```
\begin{tikzpicture}
\filldraw[fill=gray!20,draw=black,thick]
  (0,0) -- (4,0) -- (2,3) -- cycle;
\end{tikzpicture}
```

Dobija se



Opcija `fill` određuje izgled unutrašnjosti oblasti, dok opcija `draw` određuje boju njene granice. Ostale opcije, kao što je `thick`, mogu se kombinovati sa njima.

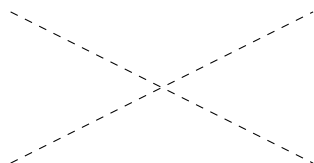
Ako se ista kombinacija opcija koristi na više mesta, može joj se dodeliti posebno ime pomoću komande `\tikzset`. Na primer,

```
\tikzset{
  pomocna/.style={dashed,thin}
}

\begin{tikzpicture}
\draw[pomocna] (0,0) -- (4,2);
\draw[pomocna] (0,2) -- (4,0);
\end{tikzpicture}
```

```
\end{tikzpicture}
```

Dobija se



U ovom primeru stil `pomocna` predstavlja kombinaciju opcija `dashed` i `thin`. Nakon njegove definicije dovoljno je navesti njegovo ime u uglastim zagrada. Ovakav pristup je koristan kada više elemenata crteža treba da ima isti izgled.

Različiti stilovi mogu se kombinovati i na istom geometrijskom crtežu. Na primer,

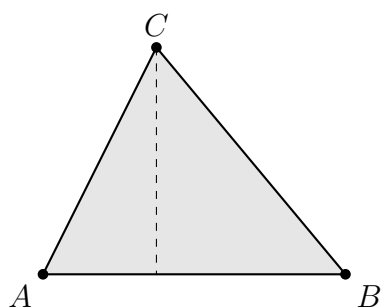
```
\begin{tikzpicture}
\coordinate (A) at (0,0);
\coordinate (B) at (4,0);
\coordinate (C) at (1.5,3);

\filldraw[fill=gray!20,draw=black,thick]
  (A) -- (B) -- (C) -- cycle;

\draw[dashed] (C) -- (1.5,0);

\fill (A) circle (2pt) node[below left] {$A$};
\fill (B) circle (2pt) node[below right] {$B$};
\fill (C) circle (2pt) node[above] {$C$};
\end{tikzpicture}
```

Dobija se



U ovom primeru stranice trougla istaknute su debljom linijom, njegova unutrašnjost je blago obojena, dok je pomoćna duž prikazana isprekidanom linijom. Izborom odgovarajućih stilova različiti elementi crteža mogu se vizuelno razlikovati prema svojoj ulozi.

Stilovi omogućavaju da se TikZ crtež učini preglednijim i da se naglase njegovi važni elementi. U narednoj podsekciji povezaćemo prethodno uvedene mogućnosti u složeniji geometrijski crtež.

4.6 Složeniji geometrijski crteži

Složeniji geometrijski crteži često zahtevaju određivanje novih tačaka na osnovu već postojećih, pronalaženje preseka različitih linija i označavanje uglova. Osnovne mogućnosti TikZ-a za takve konstrukcije mogu se proširiti odgovarajućim bibliotekama.

Biblioteke se uključuju u preambuli pomoću komande `\usetikzlibrary`. Za geometrijske crteže korisne su, između ostalih, biblioteke `calc`, `intersections`, `angles` i `quotes`. Kada se TikZ koristi zajedno sa paketom `babel`, korisno je uključiti i TikZ biblioteku `babel`, koja sprečava probleme sa pojedinim aktivnim znakovima.

```
\usetikzlibrary{calc,intersections,angles,quotes,babel}
```

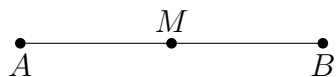
Biblioteka `calc` omogućava određivanje novih koordinata na osnovu postojećih. Na primer, sredina duži AB može se odrediti bez ručnog računanja njenih koordinata.

```
\begin{tikzpicture}
\coordinate (A) at (0,0);
\coordinate (B) at (4,0);
\coordinate (M) at ($(A)!0.5!(B)$);

\draw (A) -- (B);

\fill (A) circle (2pt) node[below] {$A$};
\fill (B) circle (2pt) node[below] {$B$};
\fill (M) circle (2pt) node[above] {$M$};
\end{tikzpicture}
```

Dobija se



Zapis `$(A)!0.5!(B)$` određuje tačku koja se nalazi na polovini puta od tačke A do tačke B . Promenom vrednosti `0.5` mogu se određivati i druge tačke na istoj duži.

Biblioteka `calc` omogućava i određivanje ortogonalne projekcije tačke na pravu. Na primer, ako su A , B i C temena trougla, podnožje visine iz temena A na pravu BC može se odrediti zapisom

```
\coordinate (D) at ($(B)!(A)!(C)$);
```

U ovom zapisu tačka D predstavlja ortogonalnu projekciju tačke A na pravu određenu tačkama B i C .

Kada je potrebno pronaći presek dve konstruisane putanje, može se koristiti biblioteka `intersections`. Putanjama se najpre dodeljuju imena, a zatim se njihov presek određuje pomoću opcije `name intersections`.

Sledeći primer povezuje ove mogućnosti u konstrukciji ortocentra trougla.

```
\begin{tikzpicture}
\coordinate (A) at (1.5,3);
\coordinate (B) at (0,0);
\coordinate (C) at (4,0);
```

```

\coordinate (D) at ($(B)!(A)!(C)$);
\coordinate (E) at ($(A)!(B)!(C)$);

\path[name path=visinaA] (A) -- (D);
\path[name path=visinaB] (B) -- (E);

\path[
  name intersections={
    of=visinaA and visinaB,
    by=H
  }
];

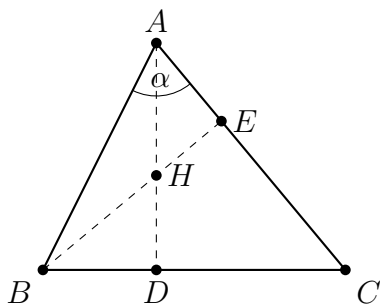
\draw[thick] (A) -- (B) -- (C) -- cycle;
\draw[dashed] (A) -- (D);
\draw[dashed] (B) -- (E);

\fill (A) circle (2pt) node[above] {$A$};
\fill (B) circle (2pt) node[below left] {$B$};
\fill (C) circle (2pt) node[below right] {$C$};
\fill (D) circle (2pt) node[below] {$D$};
\fill (E) circle (2pt) node[right] {$E$};
\fill (H) circle (2pt) node[right] {$H$};

\pic[
  draw,
  "$\alpha$",
  angle radius=7mm,
  angle eccentricity=0.7
] {angle=B--A--C};
\end{tikzpicture}

```

Dobija se



Tačke D i E predstavljaju podnožja dve visine trougla. Pomoću opcije `name path` tim visinama dodeljena su imena `visinaA` i `visinaB`, a biblioteka `intersections` određuje njihovu presečnu tačku H . Tačka H je zato ortocentar prikazanog trougla.

Biblioteke `angles` i `quotes` omogućavaju označavanje uglova. Komanda `\pic` u prethodnom primeru označava ugao BAC i pridružuje mu oznaku α . Opcija `angle radius` određuje veličinu luka kojim je ugao označen, dok `angle eccentricity` utiče na položaj njegove oznake.

Na ovom primeru mogu se videti gotovo svi osnovni elementi koje smo koristili pri radu sa TikZ-om. Crtež se gradi pomoću koordinata i putanja, pojedini elementi dobijaju odgovarajuće stilove, tačkama i geometrijskim objektima dodaju se matematičke oznake, a nove tačke mogu se konstruisati iz odnosa između već postojećih elemenata.

TikZ na taj način omogućava da se matematički crteži opišu neposredno u izvornom LaTeX dokumentu i generišu zajedno sa tekstom. Pošto se crtež zasniva na koordinatama, geometrijskim odnosima i odgovarajućim komandama, njegove pojedine delove moguće je naknadno menjati bez izrade novog grafičkog fajla. Time se dobija ujednačen način pripreme teksta, matematičkih izraza i pratećih grafičkih prikaza.

5 Prezentacije u Beamer-u

5.1 Osnovna struktura prezentacije

Beamer je LaTeX klasa namenjena izradi prezentacija. Osnovni način rada ostaje sličan pripremi drugih LaTeX dokumenata, ali se sadržaj organizuje u slajdove umesto u stranice običnog teksta.

Beamer prezentacija započinje izborom klase `beamer`.

```
\documentclass{beamer}
```

Za razliku od dokumenata koje smo ranije pravili pomoću klase `article`, ovde klasa dokumenta određuje da će sadržaj biti organizovan kao prezentacija.

U Beamer dokumentu mogu se koristiti i paketi koje smo prethodno upoznali. Na primer, za dokument na srpskom jeziku i matematički sadržaj mogu se uključiti odgovarajući paketi na isti način kao i ranije.

```
\usepackage[T1]{fontenc}
\usepackage[serbian]{babel}
\usepackage{amsmath}
\usepackage{amssymb}
```

Na isti način se po potrebi mogu uključiti paketi za slike, TikZ i druge elemente koji će biti korišćeni u prezentaciji.

Osnovni vizuelni izgled Beamer prezentacije može se odrediti izborom teme. Jedna od dostupnih tema je `Madrid`.

```
\usetheme{Madrid}
```

Tema određuje osnovni izgled slajdova, naslova i drugih elemenata prezentacije. U nastavku ćemo koristiti temu `Madrid`, dok detaljnije prilagođavanje tema nećemo razmatrati.

Po potrebi se mogu ukloniti pojedini elementi koje Beamer podrazumevano prikazuje. Na primer,

```
\setbeamertemplate{navigatation symbols}{}
```

Ova komanda uklanja navigacione simbole sa slajdova.

Osnovni podaci o prezentaciji zadaju se pomoću komandi sličnih onima koje smo koristili kod običnog LaTeX dokumenta. Pored naslova i autora, mogu se navesti podnaslov, institucija i datum.

```
\title{Značajne tačke trougla}
\subtitle{Obrazovni softver}
\author{Petar Petrović}
\institute{Matematički fakultet}
\date{}
```

Komanda `\title` određuje naslov prezentacije, a `\subtitle` njen podnaslov. Podaci o autoru i instituciji zadaju se komandama `\author` i `\institute`. Ako datum ne želimo da prikazemo, komandi `\date` možemo proslediti prazan argument.

Naslovni slajd može se napraviti pomoću komande `\titlepage`. Kao i drugi slajdovi, on se nalazi unutar okruženja `frame`.

```
\begin{frame}
\titlepage
\end{frame}
```

Na naslovnom slajdu Beamer prikazuje prethodno zadate podatke o prezentaciji. Okruženje `frame` predstavlja osnovnu jedinicu Beamer prezentacije i detaljnije ćemo ga razmotriti u narednoj podsekciji.

Jedna jednostavna i potpuna Beamer prezentacija može izgledati ovako:

```
\documentclass{beamer}

\usetheme{Madrid}
\setbeamertemplate{navigation symbols}{}
\setbeamertemplate{headline}{}

\usepackage[T1]{fontenc}
\usepackage[serbian]{babel}
\usepackage{amsmath}
\usepackage{amssymb}

\title{Značajne tačke trougla}
\subtitle{Obrazovni softver}
\author{Petar Petrović}
\institute{Matematički fakultet}
\date{}

\begin{document}

\begin{frame}
\titlepage
\end{frame}

\end{document}
```

Kompajliranjem ovog dokumenta dobija se prezentacija koja sadrži samo naslovni slajd. Njegov izgled određuje izabrana Beamer tema, dok se prikazani podaci preuzimaju iz odgovarajućih komandi zadatih u preambuli.

Osnovna struktura Beamer dokumenta zato je slična strukturi drugih LaTeX dokumenata. Najvažnija razlika je u tome što se sadržaj prezentacije organizuje u pojedinačne slajdove pomoću okruženja `frame`. U narednoj podsekciji detaljnije ćemo razmotriti pravljenje slajdova i organizaciju njihovog sadržaja.

5.2 Slajdovi i organizacija sadržaja

Osnovna jedinica Beamer prezentacije je slajd, koji se zapisuje pomoću okruženja `frame`. Sve što se nalazi unutar ovog okruženja prikazuje se na jednom slajdu prezentacije.

Najjednostavniji slajd može se zapisati ovako:

```
\begin{frame}
Ovo je sadržaj jednog slajda.
```

```
\end{frame}
```

Kompajliranjem ovog koda dobija se jedan slajd sa navedenim tekstom.

Slajd najčešće ima i naslov. On se može navesti neposredno prilikom započinjanja okruženja `frame`.

```
\begin{frame}{Osnovni pojmovi}  
Ovo je sadržaj slajda.  
\end{frame}
```

Tekst `Osnovni pojmovi` prikazuje se kao naslov slajda, dok se ostatak sadržaja nalazi ispod njega.

Naslov se može zadati i posebnom komandom `\frametitle`.

```
\begin{frame}  
\frametitle{Osnovni pojmovi}  
  
Ovo je sadržaj slajda.  
\end{frame}
```

Oba zapisa daju isti osnovni rezultat. U nastavku ćemo uglavnom koristiti kraći oblik u kome se naslov navodi neposredno uz okruženje `frame`.

Prezentacija se gradi dodavanjem više ovakvih okruženja. Svako okruženje `frame` predstavlja poseban slajd.

```
\begin{frame}{Prvi slajd}  
Sadržaj prvog slajda.  
\end{frame}  
  
\begin{frame}{Drugi slajd}  
Sadržaj drugog slajda.  
\end{frame}  
  
\begin{frame}{Treći slajd}  
Sadržaj trećeg slajda.  
\end{frame}
```

Veće prezentacije mogu se organizovati u sekcije i podsekcije pomoću komandi `\section` i `\subsection`. Njihova uloga slična je odgovarajućim komandama u običnom LaTeX dokumentu.

```
\section{Osnovni pojmovi}  
  
\begin{frame}{Definicije}  
Sadržaj slajda.  
\end{frame}  
  
\subsection{Primeri}  
  
\begin{frame}{Primer}  
Sadržaj slajda sa primerom.  
\end{frame}
```

Komande `\section` i `\subsection` organizuju slajdove u logičke celine, ali same po sebi ne stvaraju nove slajdove. Za prikaz konkretnog sadržaja i dalje je potrebno koristiti okruženje `frame`.

Na osnovu sekcija i podsekcija može se automatski napraviti sadržaj prezentacije. Za to se koristi poznata komanda `\tableofcontents`, koja se obično postavlja na poseban slajd.

```
\begin{frame}{Sadržaj}
\tableofcontents
\end{frame}
```

Beamer na ovom slajdu prikazuje strukturu prezentacije formiranu na osnovu njenih sekcija i podsekcija.

Jedna jednostavno organizovana prezentacija može izgledati ovako:

```
\documentclass{beamer}

\usetheme{Madrid}

\usepackage[T1]{fontenc}
\usepackage[serbian]{babel}

\title{Značajne tačke trougla}
\author{Petar Petrović}

\begin{document}

\begin{frame}
\titlepage
\end{frame}

\begin{frame}{Sadržaj}
\tableofcontents
\end{frame}

\section{Osnovni pojmovi}

\begin{frame}{Osnovni pojmovi}
Uvodni sadržaj prezentacije.
\end{frame}

\section{Značajne tačke trougla}

\subsection{Težište}

\begin{frame}{Težište trougla}
Sadržaj o težištu trougla.
\end{frame}

\subsection{Ortocentar}

\begin{frame}{Ortocentar trougla}
```

```
Sadržaj o ortocentru trougla.
\end{frame}

\end{document}
```

U ovom primeru naslovni slajd i sadržaj nalaze se na početku prezentacije, dok su ostali slajdovi raspoređeni u odgovarajuće sekcije i podsekcije. Na taj način i veća prezentacija može imati jasnu i preglednu strukturu.

Pri organizovanju prezentacije korisno je da svaki slajd predstavlja jednu jasno određenu sadržajnu celinu. Naslov slajda treba da ukazuje na njegovu temu, dok sekcije i podsekcije povezuju srodne slajdove u veće celine. U narednoj podsekciji razmotrićemo načine za organizovanje teksta i drugih sadržaja unutar pojedinačnog slajda.

5.3 Tekst, liste i blokovi

Sadržaj Beamer slajda može se pisati pomoću istih osnovnih LaTeX elemenata koje smo koristili u običnim dokumentima. Unutar okruženja `frame` mogu se nalaziti običan tekst, različiti oblici isticanja, liste i drugi elementi.

Na primer, delovi teksta mogu se istaći pomoću već poznatih komandi `\textbf` i `\textit`.

```
\begin{frame}{Osnovni pojmovi}
Težište je \textbf{zajednička tačka} težišnih duži trougla.

Njegov položaj može se opisati i pomoću \textit{odnosa dužina}.
\end{frame}
```

Komande za formatiranje teksta koriste se u Beamer prezentaciji na isti način kao i u drugim LaTeX dokumentima.

Kada na slajdu treba prikazati više povezanih činjenica, često se koristi okruženje `itemize`.

```
\begin{frame}{Težište trougla}
\begin{itemize}
\item Težište je presek težišnih duži.
\item Uvek se nalazi u unutrašnjosti trougla.
\item Deli svaku težišnu duž u odnosu  $(2:1)$ .
\end{itemize}
\end{frame}
```

Svaka komanda `\item` predstavlja jednu stavku liste. Beamer automatski prilagođava izgled oznaka izabranoj temi prezentacije.

Ako je redosled stavki važan, može se koristiti numerisana lista pomoću okruženja `enumerate`.

```
\begin{frame}{Konstrukcija}
\begin{enumerate}
\item Odrediti sredine stranica trougla.
\item Povezati svako teme sa sredinom naspramne stranice.
\item Odrediti presek dobijenih težišnih duži.
\end{enumerate}
\end{frame}
```

Beamer automatski numerise stavke i prikazuje ih u skladu sa izabranom temom.

Liste mogu biti i ugnježdene kada je potrebno prikazati hijerarhiju sadržaja.

```
\begin{frame}{Položaj ortocentra}
\begin{itemize}
\item Položaj ortocentra zavisi od vrste trougla.
  \begin{itemize}
    \item kod oštroglog je u unutrašnjosti,
    \item kod pravouglog je u temenu pravog ugla,
    \item kod tupouglog je izvan trougla.
  \end{itemize}
\end{itemize}
\end{frame}
```

Ugnježdene liste treba koristiti umereno, kako bi struktura slajda ostala jednostavna i pregledna.

Beamer sadrži i posebna okruženja za vizuelno izdvajanje sadržaja. Najčešće se koristi okruženje `block`, koje ima sopstveni naslov.

```
\begin{frame}{Ojlerova prava}
\begin{block}{Napomena}
Centar opisane kružnice, težište i ortocentar nalaze se na istoj pravoj.
\end{block}
\end{frame}
```

Naslov bloka navodi se kao argument okruženja `block`, dok se njegov sadržaj zapisuje unutar samog okruženja. Konačan izgled bloka zavisi od izabrane Beamer teme.

Za izdvajanje primera može se koristiti okruženje `exampleblock`.

```
\begin{frame}{Primer}
\begin{exampleblock}{Jednakostranični trougao}
Kod jednakostraničnog trougla težište, ortocentar, centar opisane i centar upisane
  kružnice predstavljaju istu tačku.
\end{exampleblock}
\end{frame}
```

Slično tome, posebno važan sadržaj ili upozorenje može se istaći pomoću okruženja `alertblock`.

```
\begin{frame}{Važna napomena}
\begin{alertblock}{Obratiti pažnju}
Ortocentar ne mora da se nalazi u unutrašnjosti trougla.
\end{alertblock}
\end{frame}
```

Okruženja `block`, `exampleblock` i `alertblock` imaju sličnu strukturu, ali ih Beamer vizuelno razlikuje. Njihova upotreba omogućava da se pojedini delovi slajda istaknu prema svojoj ulozi.

Različiti elementi mogu se kombinovati na istom slajdu. Na primer,

```
\begin{frame}{Ortocentar trougla}

Ortocentar  $(H)$  predstavlja presek pravih koje sadrže visine trougla.

\begin{itemize}
```

```

\item Njegov položaj zavisi od vrste trougla.
\item Kod oštroglog trougla nalazi se u unutrašnjosti.
\item Kod tupouglog trougla nalazi se izvan trougla.
\end{itemize}

\begin{block}{Poseban slučaj}
Kod pravouglog trougla ortocentar se nalazi u temenu pravog ugla.
\end{block}

\end{frame}

```

Ovakva organizacija omogućava da se osnovna tvrdnja kratko uvede tekstom, važne činjenice prikazu u obliku liste, a poseban zaključak dodatno izdvoji blokom.

Pri pripremi slajdova važno je da količina teksta ostane ograničena i da se vizuelno izdvajanje koristi samo kada doprinosi preglednosti sadržaja. U narednoj podsekciji razmotrićemo prikaz matematičkog i grafičkog sadržaja u Beamer prezentaciji.

5.4 Matematički i grafički sadržaj

Matematički sadržaj u Beamer prezentaciji zapisuje se na isti način kao u drugim LaTeX dokumentima. Mogu se koristiti matematički izrazi unutar teksta, izdvojene formule i odgovarajuća matematička okruženja.

Na primer, slajd može sadržati matematički izraz u tekstu i izdvojenju formulu.

```

\begin{frame}{Težište trougla}
Težište  $(T)$  deli svaku težišnu duž u odnosu  $(2:1)$ .

Ako je  $(M)$  sredina stranice  $(BC)$ , važi
\[
AT = \frac{2}{3} AM.
\]
\end{frame}

```

Matematički režim, razlomci, indeksi, simboli i druge konstrukcije koriste se na isti način kao u običnom LaTeX dokumentu. Pri izradi prezentacije zato nije potrebno učiti poseban način zapisivanja matematičkih izraza.

Kada je potrebno prikazati nekoliko povezanih koraka, mogu se koristiti i poznata matematička okruženja. Na primer,

```

\begin{frame}{Razvoj izraza}
\begin{align*}
(a+b)^2
&= (a+b)(a+b) \\
&= a^2+2ab+b^2.
\end{align*}
\end{frame}

```

Okruženje `align*` omogućava poravnanje više redova bez njihove numeracije. U prezentacijama je često korisno prikazati samo one matematičke korake koji su neposredno važni za objašnjenje.

Beamer slajdovi mogu sadržati i postojeće grafičke fajlove. Za njihovo uključivanje može se koristiti paket `graphicx`, koji smo već upoznali.

```
\usepackage{graphicx}
```

Slika se zatim može postaviti na slajd pomoću komande `\includegraphics`.

```
\begin{frame}{Geometrijski prikaz}  
\centering  
\includegraphics[width=0.7\textwidth]{trougao.png}  
\end{frame}
```

Opcija `width` određuje širinu slike u odnosu na širinu prostora namenjenog sadržaju. Komanda `\centering` postavlja sliku na sredinu slajda.

TikZ crteži mogu se uključiti neposredno u Beamer slajd. Potrebno je da paket `tikz` bude uključen u preambuli prezentacije.

```
\usepackage{tikz}
```

Nakon toga se okruženje `tikzpicture` koristi na isti način kao u običnom LaTeX dokumentu.

```
\begin{frame}{Trougao}  
\centering  
  
\begin{tikzpicture}  
\coordinate (A) at (0,3);  
\coordinate (B) at (-2,0);  
\coordinate (C) at (2,0);  
  
\draw[thick] (A) -- (B) -- (C) -- cycle;  
  
\fill (A) circle (2pt) node[above] {$A$};  
\fill (B) circle (2pt) node[below left] {$B$};  
\fill (C) circle (2pt) node[below right] {$C$};  
\end{tikzpicture}  
\end{frame}
```

Na ovaj način matematički crtež postaje neposredan deo prezentacije i može se menjati pomoću istih TikZ komandi koje smo prethodno upoznali.

Na jednom slajdu često je korisno rasporediti sadržaj u više delova. Beamer za to sadrži okruženje `columns`, unutar koga se pojedinačne kolone zadaju pomoću okruženja `column`.

```
\begin{frame}{Težište trougla}  
\begin{columns}  
  
\begin{column}{0.48\textwidth}  
Težište je presek težišnih duži trougla.  
  
\begin{itemize}  
\item nalazi se u unutrašnjosti trougla,  
\item deli svaku težišnu duž u odnosu  $(2:1)$ .  
\end{itemize}  
\end{column}  
\end{columns}
```

```

\begin{column}{0.48\textwidth}
\centering

\begin{tikzpicture}[scale=0.8]
\coordinate (A) at (0,3);
\coordinate (B) at (-2,0);
\coordinate (C) at (2,0);

\coordinate (M) at (0,0);
\coordinate (N) at (1,1.5);
\coordinate (P) at (-1,1.5);

\draw[thick] (A) -- (B) -- (C) -- cycle;
\draw[dashed] (A) -- (M);
\draw[dashed] (B) -- (N);
\draw[dashed] (C) -- (P);

\fill (A) circle (2pt) node[above] {$A$};
\fill (B) circle (2pt) node[below left] {$B$};
\fill (C) circle (2pt) node[below right] {$C$};
\fill (0,1) circle (2pt) node[right] {$T$};
\end{tikzpicture}
\end{column}

\end{columns}
\end{frame}

```

Okruženje `columns` određuje deo slajda koji će biti podeljen na kolone, dok svako okruženje `column` predstavlja jednu od njih. U ovom primeru obe kolone imaju širinu `0.48\textwidth`, pa je u levoj smešten tekst, a u desnoj TikZ crtež.

Kombinovanje teksta, matematičkih izraza i grafičkih prikaza omogućava da se sadržaj slajda organizuje u skladu sa njegovom namenom. Elemente koji su već poznati iz običnih LaTeX dokumenata nije potrebno zapisivati na poseban način, dok Beamer pruža dodatne mogućnosti, kao što su kolone, za njihovo raspoređivanje na slajdu. U narednoj podsekciji razmotrićemo postepeno prikazivanje sadržaja tokom prezentacije.

5.5 Postepeno prikazivanje sadržaja

Tokom izlaganja često nije potrebno da se čitav sadržaj slajda prikaže odjednom. Beamer omogućava da se pojedini elementi pojavljuju postepeno, čime se pažnja može usmeriti na deo sadržaja koji se u tom trenutku objašnjava.

Najjednostavniji način za postepeno prikazivanje sadržaja je komanda `\pause`. Sadržaj koji se nalazi posle ove komande pojavljuje se u narednom koraku prikazivanja slajda.

```

\begin{frame}{Težište trougla}
Težište je presek težišnih duži trougla.

\pause

Uvek se nalazi u unutrašnjosti trougla.

```

```
\pause
```

Svaku težišnu duž deli u odnosu $\backslash(2:1\backslash)$.

```
\end{frame}
```

Pri prvom prikazu slajda vidljiva je samo prva tvrdnja. Nakon prelaska na sledeći korak pojavljuje se druga, a zatim i treća tvrdnja.

Komanda `\pause` naročito je pogodna za postepeno prikazivanje stavki liste.

```
\begin{frame}{Konstrukcija težišta}
```

```
\begin{enumerate}
```

```
\item Odrediti sredine stranica trougla.
```

```
\pause
```

```
\item Povezati temena sa sredinama naspramnih stranica.
```

```
\pause
```

```
\item Odrediti presek težišnih duži.
```

```
\end{enumerate}
```

```
\end{frame}
```

Stavke se u ovom primeru pojavljuju jedna za drugom tokom prikazivanja slajda.

Beamer ovakvo postepeno prikazivanje ostvaruje tako što generiše više uzastopnih PDF stranica koje predstavljaju različita stanja istog slajda. Tokom prezentacije prelazak između tih stranica ostavlja utisak postepenog pojavljivanja sadržaja.

Kada je potrebna preciznija kontrola, mogu se koristiti takozvane overlay specifikacije. One određuju u kojim koracima određeni element treba da bude prikazan. Na primer,

```
\begin{frame}{Položaj ortocentra}
```

```
\begin{itemize}
```

```
\item<1-> Kod oštroglog trougla nalazi se u unutrašnjosti.
```

```
\item<2-> Kod pravouglog trougla nalazi se u temenu pravog ugla.
```

```
\item<3-> Kod tupouglog trougla nalazi se izvan trougla.
```

```
\end{itemize}
```

```
\end{frame}
```

Zapis `<1->` znači da je prva stavka vidljiva od prvog koraka nadalje. Druga stavka, označena sa `<2->`, pojavljuje se od drugog koraka, dok se treća pojavljuje od trećeg koraka.

Za jednostavne prezentacije komanda `\pause` često je sasvim dovoljna. Overlay specifikacije su korisne kada je potrebno preciznije odrediti redosled pojavljivanja pojedinih elemenata.

Na primer, postupak određivanja težišta može se prikazivati postepeno ovako:

```
\begin{frame}{Određivanje težišta}
```

Težište trougla određuje se pomoću njegovih težišnih duži.

```
\begin{enumerate}
```

```
\item<1-> Odrede se sredine stranica trougla.
```

```
\item<2-> Svako teme poveže se sa sredinom naspramne stranice.
```

```
\item<3-> Dobijene težišne duži seku se u jednoj tački.
```

```
\end{enumerate}
```

```
\begin{block}<4->{Zaključak}
```

```
Njihova zajednička tačka je težište trougla.  
\end{block}  
  
\end{frame}
```

U ovom primeru prve tri faze prikazuju se redom, dok se zaključak pojavljuje tek u četvrtom koraku. Overlay specifikacija može se, dakle, koristiti ne samo uz stavke liste već i uz druge Beamer elemente, kao što je blok.

Postepeno prikazivanje sadržaja treba koristiti kada ono prati tok objašnjenja i doprinosi preglednosti slajda. Ako nema potrebe da se elementi pojavljuju odvojeno, jednostavnije je prikazati ceo sadržaj odmah.

Beamer omogućava da se LaTeX koristi za izradu jasno organizovanih prezentacija sa naslovnim slajdom, sekcijama, listama, blokovima, matematičkim izrazima i grafičkim prikazima. Postojeće LaTeX znanje može se neposredno primeniti, dok posebne mogućnosti kao što su kolone i postepeno prikazivanje sadržaja omogućavaju njegovo prilagođavanje prezentacijskom formatu. Nakon pripreme dokumenata i prezentacija u LaTeX-u, u narednoj sekciji prelazimo na tehnologije koje omogućavaju prikaz matematičkog i obrazovnog sadržaja u veb-okruženju.

6 Veb-tehnologije

6.1 Bootstrap

Bootstrap je biblioteka koja omogućava jednostavnije oblikovanje veb-stranica pomoću unapred pripremljenih CSS stilova. Umesto da se izgled svakog elementa stranice definiše od početka, HTML elementima mogu se dodeliti odgovarajuće Bootstrap klase i na taj način koristiti već pripremljena pravila za njihov prikaz.

Bootstrap omogućava jednostavno podešavanje rasporeda sadržaja, veličine i položaja elemenata, razmaka, boja, dugmadi, navigacije, tabela, obrazaca i drugih često korišćenih delova veb-stranice. Posebno je korisno što olakšava izradu stranica koje se prilagođavaju različitim veličinama ekrana.

Za korišćenje Bootstrap-a potrebno je povezati odgovarajući Bootstrap CSS fajl sa HTML dokumentom. Nakon toga se njegove mogućnosti koriste dodavanjem odgovarajućih klasa HTML elementima. Na taj način osnovni izgled stranice može se napraviti bez pisanja velike količine sopstvenog CSS koda.

U nastavku ćemo podrazumevati korišćenje Bootstrapa 5. Nije potrebno poznavati sve njegove mogućnosti, već je dovoljno upoznati one klase koje su potrebne za organizovanje i oblikovanje konkretne veb-stranice.

Osnovne mogućnosti Bootstrapa 5, zajedno sa brojnim primerima njihove upotrebe, mogu se upoznati u tutorijalu na adresi <https://www.w3schools.com/bootstrap5/>. Zvanična dokumentacija dostupna je na adresi <https://getbootstrap.com/>.

Bootstrap tako pruža jednostavan način za oblikovanje i organizovanje sadržaja veb-stranice korišćenjem već pripremljenih stilova. U narednoj podsekciji upoznaćemo MathJax, koji omogućava prikaz matematičkih izraza u veb-okruženju.

6.2 MathJax

MathJax je biblioteka koja omogućava prikaz matematičkih izraza neposredno u veb-stranicama. Matematički sadržaj zapisuje se tekstualno, a MathJax ga u veb-pregledaču pretvara u pravilno oblikovane formule.

MathJax podržava više načina zapisivanja matematičkog sadržaja, među kojima su LaTeX, MathML i AsciiMath. U nastavku ćemo koristiti LaTeX zapis, pošto smo njegovu sintaksu već upoznali. MathJax pri tome ne obrađuje čitav LaTeX dokument, već matematičke izraze koji se nalaze unutar HTML sadržaja.

Najjednostavniji način korišćenja MathJax-a jeste njegovo učitavanje sa udaljenog servera. U zaglavlje HTML dokumenta može se dodati sledeća naredba:

```
<script defer
  src="https://cdn.jsdelivr.net/npm/mathjax@4/tex-ctml.js">
</script>
```

Nakon učitavanja MathJax-a matematički izrazi mogu se zapisivati neposredno unutar HTML sadržaja. Za matematiku unutar teksta podrazumevano se koriste oznake $(\dots)$, dok se za izdvojene formule mogu koristiti oznake $[\dots]$.

Jednostavan HTML dokument sa matematičkim sadržajem može izgledati ovako:

```
<!doctype html>
<html lang="sr">
<head>
```

```

<meta charset="utf-8">
<title>MathJax primer</title>

<script defer
  src="https://cdn.jsdelivr.net/npm/mathjax@4/tex-ctml.js">
</script>
</head>
<body>

<p>
Ako je  $(a=3)$  i  $(b=4)$ , onda je
 $(\sqrt{a^2+b^2}=5)$ .
</p>

<p>Važi i formula</p>


$$(a+b)^2=a^2+2ab+b^2.$$


</body>
</html>

```

MathJax pronalazi matematičke izraze u HTML dokumentu i prikazuje ih kao odgovarajuće matematičke formule. Sintaksa samih izraza uglavnom je ista kao ona koju smo koristili pri pisanju matematičkog sadržaja u LaTeX-u.

Za izdvojene matematičke izraze MathJax podrazumevano prepoznaje i zapis $$$\dots$$$. Sa druge strane, zapis $\$ \dots \$$ za matematiku unutar teksta nije podrazumevano uključen. Ako želimo da koristimo i ovaj poznati način zapisivanja, MathJax se može dodatno podesiti pre njegovog učitavanja.

```

<script>
window.MathJax = {
  tex: {
    inlineMath: ['+', '']: [['$', '$']]
  }
};
</script>

<script defer
  src="https://cdn.jsdelivr.net/npm/mathjax@4/tex-ctml.js">
</script>

```

Nakon ovog podešavanja matematika unutar teksta može se zapisivati i između znakova $\$$, kao u običnom LaTeX dokumentu.

MathJax se ne mora učitavati preko udaljenog servera. Njegovi fajlovi mogu se čuvati i zajedno sa ostalim fajlovima veb-sajta ili na sopstvenom serveru. U tom slučaju u elementu `script` navodi se odgovarajuća lokalna putanja.

```

<script defer src="mathjax/tex-ctml.js"></script>

```

Ovakav način može biti koristan kada je potrebno da stranica radi bez pristupa internetu. Za uobičajenu upotrebu jednostavnije je učitavanje MathJax-a preko udaljenog servera.

Detaljnije informacije o pisanju matematičkog sadržaja pomoću MathJax-a dostupne su na adresi <https://docs.mathjax.org/en/latest/basic/mathematics.html>.

MathJax tako omogućava da se matematički izrazi zapisani poznatom LaTeX sintaksom prikažu neposredno unutar veb-stranice. U narednoj podsekciji razmotrićemo GeoGebru, koja omogućava uključivanje interaktivnih matematičkih sadržaja u veb-okruženje.

6.3 GeoGebra

GeoGebra je program za izradu interaktivnih matematičkih sadržaja. Omogućava rad sa geometrijskim konstrukcijama, funkcijama, jednačinama i drugim matematičkim objektima, pri čemu korisnik može neposredno da menja pojedine elemente i posmatra kako te promene utiču na ostatak konstrukcije.

GeoGebra konstrukcija može se sačuvati u fajlu sa ekstenzijom `.ggb`. Način izrade ovakvih konstrukcija obrađuje se na vežbama, dok ćemo ovde razmotriti samo način njihovog uključivanja u HTML stranicu.

Za prikaz GeoGebra sadržaja na veb-stranici potrebno je najpre učitati odgovarajuću GeoGebra biblioteku. To se može uraditi dodavanjem sledećeg elementa u HTML dokument.

```
<script src="https://www.geogebra.org/apps/deployggb.js"></script>
```

Na mestu na kome želimo da se aplet pojavi potrebno je napraviti odgovarajući HTML element.

```
<div id="ggb-element"></div>
```

Nakon toga se kreira GeoGebra aplet i povezuje sa postojećim `.ggb` fajlom.

```
<script>
var params = {
  filename: "aplet.ggb",
  width: 800,
  height: 600
};

var applet = new GGBApplet(params, true);

window.addEventListener("load", function() {
  applet.inject("ggb-element");
});
</script>
```

Parametar `filename` određuje GeoGebra fajl koji treba učitati, dok parametri `width` i `height` određuju dimenzije prikazanog apleta. Komanda `inject` smešta aplet u HTML element čiji je identifikator `ggb-element`.

Ceo jednostavan HTML dokument može zato izgledati ovako:

```
<!doctype html>
<html lang="sr">
<head>
  <meta charset="utf-8">
```

```

<meta name="viewport"
      content="width=device-width, initial-scale=1">

<title>GeoGebra primer</title>

<script src="https://www.geogebra.org/apps/deployggb.js"></script>
</head>
<body>

<h1>Interaktivna konstrukcija</h1>

<div id="ggb-element"></div>

<script>
var params = {
  filename: "aplet.ggb",
  width: 800,
  height: 600
};

var applet = new GGBApplet(params, true);

window.addEventListener("load", function() {
  applet.inject("ggb-element");
});
</script>

</body>
</html>

```

U ovom primeru pretpostavlja se da se fajl `aplet.ggb` nalazi na odgovarajućem mestu u odnosu na HTML dokument. Nakon učitavanja stranice GeoGebra prikazuje sadržaj tog fajla unutar elementa `ggb-element`.

GeoGebra aplet može se učitati i na osnovu materijala koji je prethodno objavljen na GeoGebra sajtu. U tom slučaju umesto parametra `filename` može se koristiti odgovarajući parametar `material_id`. Za potrebe ove skripte dovoljno je koristiti prikazani način rada sa postojećim `.ggb` fajlom.

Detaljnije informacije o uključivanju GeoGebra sadržaja u veb-stranice dostupne su na adresi https://geogebra.github.io/docs/reference/en/GeoGebra_Apps_Embedding/.

Bootstrap, MathJax i GeoGebra imaju različite uloge pri izradi veb-stranice. Bootstrap omogućava jednostavnije oblikovanje i organizovanje sadržaja, MathJax prikaz matematičkih izraza, a GeoGebra uključivanje interaktivnih matematičkih konstrukcija. Njihovom kombinovanom upotrebom može se napraviti pregledna veb-stranica koja sadrži tekstualni, matematički i interaktivni sadržaj.

7 Projekat iz Obrazovnog softvera

7.1 Cilj projekta

Projekat iz Obrazovnog softvera predstavlja obradu jedne matematičke teme korišćenjem više tehnologija za pripremu obrazovnih materijala. Ista tema obrađuje se kroz seminarski rad, prezentaciju, interaktivni GeoGebra aplet i obrazovni veb-sajt, pri čemu svaki deo treba prilagoditi načinu na koji se sadržaj predstavlja.

U okviru projekta koriste se LaTeX, TikZ, Beamer, GeoGebra, Bootstrap i MathJax. Tehnologije koje smo upoznali u prethodnim sekcijama na taj način se povezuju u jednu celinu.

7.2 Prijava teme i bodovanje

Tema projekta treba da bude matematička i dovoljno sadržajna da može smisleno da se obradi kroz sve njegove delove. Prijava teme šalje se mejlom i treba da sadrži ime i prezime studenta i predloženi naziv teme.

Seminarski rad izrađen u LaTeX-u nosi 10 poena. Beamer prezentacija, GeoGebra aplet i odbrana zajedno nose 20 poena. Obrazovni veb-sajt nosi 10 poena.

7.3 Celine projekta

Projekat se sastoji od četiri povezane celine koje obrađuju istu izabranu matematičku temu na različite načine.

Seminarski rad predstavlja detaljniju pisanu obradu izabrane teme i izrađuje se u LaTeX-u. Potrebno je da sadrži naslovnu stranu, sadržaj, smisleno organizovane sekcije i spisak literature. U radu treba koristiti matematičke izraze, primere i grafičke prikaze u skladu sa temom. Obavezna je najmanje jedna smisljena TikZ ilustracija, a mogu se koristiti i druge slike ili grafički materijali.

Beamer prezentacija treba da bude sažeta i prilagođena usmenom izlaganju, a ne kopija seminarskog rada. Očekuje se uredna struktura, matematički izrazi i odgovarajući grafički elementi, uz smisleno korišćenje mogućnosti Beamer-a.

GeoGebra aplet mora biti interaktivan i povezan sa matematičkim sadržajem projekta. Nije dovoljno napraviti samo statičan crtež. Interaktivnost može, u zavisnosti od teme, da obuhvati pokretne ili zavisne objekte, klizače, kontrole za prikazivanje pojedinih elemenata ili druge odgovarajuće mogućnosti GeoGebre.

Obrazovni veb-sajt izrađuje se pomoću Bootstrapa i treba da ima jasnu strukturu, više međusobno povezanih stranica, zajedničku navigaciju, prilagođen prikaz na različitim veličinama ekrana i ujednačen vizuelni stil. Bootstrap komponente treba koristiti tamo gde odgovaraju sadržaju. Za prikaz matematičkih izraza koristi se MathJax. JavaScript se može koristiti po potrebi, ali nije obavezan. Na sajtu treba da budu dostupne PDF verzije seminarskog rada i prezentacije, kao i ugrađen GeoGebra aplet.

7.4 Predaja i odbrana

Kompletan projekat šalje se mejlom do 31. avgusta 2027. Uz PDF verzije seminarskog rada i prezentacije potrebno je priložiti i izvorne `.tex` fajlove, slike i druge resurse potrebne za njihovo kompajliranje, kompletan veb-sajt sa pratećim fajlovima i GeoGebra `.ggb` fajl.

Na odbrani se izabrana tema predstavlja pomoću izrađene Beamer prezentacije i demonstrira se GeoGebra aplet. Ostale delove projekta nije potrebno prezentovati niti moraju biti završeni do termina odbrane.

7.5 Primer projekta

Kao primer načina na koji se jedna matematička tema može povezati kroz sve delove projekta dostupan je projekat *Značajne tačke trougla* na adresi <https://poincare.matf.bg.ac.rs/~stefan.miskovic/os/projekat/>.

Primer služi kao orijentacija za očekivani tip i obim projekta, a ne kao šablon koji treba doslovno pratiti.