

# LATEX program za oblikovanje dokumenata i izrada LATEX predložka za diplomski rad

---

**Milin, Matea**

**Master's thesis / Diplomski rad**

**2019**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Graphic Arts / Sveučilište u Zagrebu, Grafički fakultet**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:216:023225>

*Rights / Prava:* [In copyright](#) / [Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-08-31**



*Repository / Repozitorij:*

[Faculty of Graphic Arts Repository](#)



Sveučilište u Zagrebu  
Grafički fakultet

Matea Milin

L<sup>A</sup>T<sub>E</sub>X PROGRAM ZA OBLIKOVANJE  
DOKUMENATA I IZRADA L<sup>A</sup>T<sub>E</sub>X  
PREDLOŠKA ZA DIPLOMSKI RAD

Diplomski rad

Zagreb, 2019.



Sveučilište u Zagrebu  
Grafički fakultet

Matea Milin

L<sup>A</sup>T<sub>E</sub>X PROGRAM ZA OBLIKOVANJE  
DOKUMENATA I IZRADA L<sup>A</sup>T<sub>E</sub>X  
PREDLOŠKA ZA DIPLOMSKI RAD

Diplomski rad

Mentor:

Izv. dr. sc. Ante Poljičak

Student:

Matea Milin

Zagreb, 2019.

Rješenje o odobrenju teme diplomskog rada

# Sadržaj

<b>1</b>	<b>UVOD</b>	<b>1</b>
<b>2</b>	<b>TEORIJSKI DIO</b>	<b>3</b>
2.1	TeX i povijest TeX-a . . . . .	3
2.2	Donald E. Knuth . . . . .	3
2.3	L <sup>A</sup> TeX i povijest L <sup>A</sup> TeX-a . . . . .	5
2.4	Ime TeX i L <sup>A</sup> TeX . . . . .	6
2.5	CTAN . . . . .	6
2.6	Razlika između klase i paketa . . . . .	7
2.7	Prednosti i mane (La)TeX-a . . . . .	8
2.8	Princip rada TeX-a . . . . .	9
2.9	Struktura L <sup>A</sup> TeX-a . . . . .	15
<b>3</b>	<b>EKSPERIMENTALNI DIO</b>	<b>16</b>
3.1	Postavljanje klase . . . . .	17
3.2	Izrada naslovnica . . . . .	18
3.2.1	Dimenzije i margine . . . . .	19
3.2.2	Ubacivanje UniZg fonta . . . . .	21
3.2.3	Definiranje tokena i novih naredbi . . . . .	22
3.2.4	Oblikovanje naslovne strane i prve stranice . . . . .	23
3.3	Oblikovanje teksta dokumenta . . . . .	28
3.4	Ubacivanje literature . . . . .	32
3.5	Pripremanje <i>unizg_klasa.cls</i> za postavljanje na CTAN i GitHub . . . . .	34
3.6	Korištenje <i>unizg_klasa.cls</i> ? . . . . .	35
<b>4</b>	<b>REZULTATI I RASPRAVA</b>	<b>37</b>
<b>5</b>	<b>ZAKLJUČAK</b>	<b>38</b>
<b>6</b>	<b>Literatura</b>	<b>39</b>

<b>A</b>	<b>DODATAK</b>	<b>41</b>
A.1	Kratke upute za korištenje $\LaTeX$ -a . . . . .	41
A.2	<i>Download</i> i instalacija . . . . .	41
A.3	$\LaTeX$ distribucije, <i>engines</i> i editori . . . . .	44
A.4	Početak rada . . . . .	46
	A.4.1    Upisivanje naredbi i komentari . . . . .	46
	A.4.2    Pozivanje klase . . . . .	47
A.5	Korištenje paketa . . . . .	50
A.6	Uređivanje dokumenta i osnovne naredbe . . . . .	50
	A.6.1    Postavljanje margina i dimenzije . . . . .	50
	A.6.2    Jezik i gramatika . . . . .	51
	A.6.3    Naslovi i podnaslovi . . . . .	51
	A.6.4    Naglašavanje teksta . . . . .	51
A.7	Ubacivanje slika i Graphicx paket . . . . .	51
	A.7.1    Pisanje formula i matematički mod . . . . .	53
	A.7.2    Tablice i liste . . . . .	55
	A.7.3    Citiranje i literatura . . . . .	58

# Sažetak

U ovom diplomskom radu govori se o programu za obradu tekstualnih dokumenata  $\LaTeX$ -u i njegovom računalnom mehanizmu  $\TeX$ -u. Dizajn igra važnu ulogu u prezentaciji rada, proizvoda ili samoga sebe pa tako i nekog znanstvenog rada. Osim dobre teme i sadržaja važno je dobro oblikovati tekst da čitatelju čitanje bude što lakše i ugodnije. Autori znanstvenih radova, knjiga ili članaka često nemaju vremena za estetiku, a nerijetko oblikovanje može biti frustrirajuće i teško. Iz toga proizlazi motivacija za ovaj rad i izradu  $\LaTeX$  klase koja će oblikovanje teksta pojednostavniti.

Cilj je približiti program  $\LaTeX$  studentima Grafičkog fakulteta te izraditi  $\LaTeX$  predložak ili klasu koja će olakšati studentima Sveučilišta u Zagrebu oblikovanje njihovih radova. U teorijskom dijelu ovog rada govorit će se o povijesti i nastajanju  $\LaTeX$ -a te o principu rada  $\TeX$ -a.

U praktičnom dijelu ovog rada istražiti će se način izrade  $\LaTeX$  predloška za oblikovanje teksta. Pokušat će se riješiti problem neusklađenosti diplomskih radova na Grafičkom fakultetu i opisat će se problemi na koje se naišlo pri izradi klase. Korišten je  $\LaTeX$  mark-up jezik za oblikovanje dokumenata te su praćene upute za izradu diplomskih radova grafičkog fakulteta. Krajnji rezultat je .cls datoteka koja potpuno odgovara uputama za oblikovanje diplomskog rada navedenim na mrežnim stranicama fakulteta te je predložak postavljen u internetsku bazu kako bi bila dostupna svim  $\LaTeX$  korisnicima. Na kraju rada zaključuje se da je  $\LaTeX$  sistem za oblikovanje dokumenata idealno rješenje za uređivanje dugih ili matematičkih dokumenata te da je dovoljno jednostavan za osobe bez pozadine u programiranju.

**Ključne riječi:** LaTeX, TeX, klasa, predložak, CTAN

# Abstract

This thesis deals with typesetting software  $\LaTeX$  and its engine  $\TeX$ . Design plays an important role in the presentation of a work, a product or itself, and same applies to scientific work. In addition to good subject and content, it is important to have a well typesetted text so that readers can read as easily and comfortably as possible. Authors of scientific papers, books, or articles often lack the time for aesthetics, and designing can often be frustrating and difficult. This gives a motivation for this work and creating a  $\LaTeX$  class that will simplify text formatting.

The goal is to bring  $\LaTeX$  program to students of the Faculty of Graphic Arts and create a  $\LaTeX$  template that will make easier for students from the University of Zagreb to design their papers. In the theoretical part of this paper we will talk about the history and emerging of  $\LaTeX$  and about the principle of  $\TeX$ .

In the practical part of this paper, we will explore the way to create the  $\LaTeX$  text formatting template. The problem of mismatching graduate theses at the Faculty of Graphic Arts will be attempted to solve and the encountered problems will be described. The instructions for tekst formatting provided by Faculty of Graphic Arts were followed and used language for creation of template was  $\LaTeX$  markup language. The final result is a .cls file that fully complies with the design instructions for the graduate thesis listed on the faculty's web pages and the template is uploaded to an internet database for it to be available to all  $\LaTeX$  users. At the end of the work it is concluded that the  $\LaTeX$  typesetting system is an ideal solution for editing long or mathematical documents and is easy enough for people without background in programming.

**Keywords:** LaTeX, TeX, class, template, CTAN



# 1. UVOD

Kod pisanja knjiga ili dugih članaka i izvješća često se zanemari struktura i oblikovanje teksta. Takav pristup rezultira teško čitljivim dokumentom koji umara oči i čitatelja. Naravno ni jedan autor ne želi da se njegov trud i rad izgubi u banalnom propustu kao što je uređivanje teksta. Zato se često gubi vrijeme tražeći najbolje dizajniran predložak, pokušavajući ubaciti zanimljive slike ili jednostavno tražeći na internetu kako se numeriraju stranice u nekom od programa za obradu teksta kao što je Microsoft Word. Takvi problemi su nepresušan izvor frustracija, ali i gubitak dragocjenog vremena koje se moglo utrošiti u istraživanje ili pisanje rada. Također, slaganje slogova odnosno prelamanje teksta nije jednostavan zadatak i zahtijeva stručno znanje grafičkog urednika. *What you see is what you get* (WYSIWYG) programi kao što je Microsoft Word i Apple Pages pružaju golem izbor načina za prijelom teksta, ubacivanje slika, dodavanja raznog sadržaja kojem korisnik vizualno određuje dimenzije, položaj, poravnanje i sl. Takav intuitivan pristup tim programima doveo je velik broj korisnika pa danas u svijetu skoro svi koji posjeduju računalo posjeduju i neki od tih programa. Iako je takav način pisanja za prosječnog korisnika dovoljan, kada u priču dolaze komplicirane tekstualne strukture kao što su matematičke formule, glazbene partiture, programski kodovi, ..., stvari se zakompliciraju. U tu svrhu stvoren je  $\LaTeX$ .

Motivacija za ovaj rad dolazi iz osobnog iskustva oblikovanja dugačkih dokumenata te želje da se  $\LaTeX$  približi široj publici sa sličnim problemom. Jednostavnost ovog slovoslagarskog programa uvelike bi pomogla hrvatskoj akademskoj zajednici koja s njim nije dovoljno dobro upoznata. Kao dodatna motivacija bio je i zaključak da akademska zajednica nije usklađena po pitanju oblikovanja završnih, diplomskih i doktorskih radova i disertacija već svako sveučilište pa i svaki fakultet ima svoj predložak oblikovanja teksta. Nerijetko se i unutar fakulteta nađu radovi koji potpuno odstupaju od zadanog. Iz želje da se ukaže na taj problem odlučeno je izraditi vlastiti  $\LaTeX$  predložak koji bi svaki student mogao koristiti u pisanju svog stručnog rada. Za zadatak je uzeta izrada jedinstvene klase koja će se zvati *unizg\_klasa* te će pratiti zahtjeve za oblikovanje diplomskih radova Grafičkog fakulteta u Zagrebu. Neki od zahtjeva su korištenje UniZg fonta, rad struk-

turiran na poglavlja i potpoglavlja, numeracija u donjem desnom kutu, citirani tekst i sl. Klasa će omogućavati automatsko numeriranje stranica s desne strane, uniformirane naslovne stranice, jednostavanu izradu popisa slika i tablica, jednostavno citiranje itd. Izradit će se uz pomoću  $\text{\LaTeX}$  markup jezika.

## 2. TEORIJSKI DIO

### 2.1. $\TeX$ i povijest $\TeX$ -a

$\TeX$  je besplatan slovoslagarski sustav ili sustav za oblikovanje teksta koji je napisao Donald Knuth 1978. godine.  $\TeX$  je popularno sredstvo za slaganje složenih matematičkih formula te je iznimno popularan u akademskim krugovima, matematici, računarstvu, ekonomiji, fizici itd. Cilj  $\TeX$ -a bio je da omogući svima izradu visokokvalitetnih knjiga te da rezultatu budu isti na svim računalima. Danas se  $\TeX$  najčešće koristi kao pogonski program za makro pakete kao što su  $\LaTeX$ , ConTeXt, LuaLaTeX, XeLaTeX itd.

Donald Knuth, autor  $\TeX$ -a, 1968. godine objavljuje rad *The Art of Computer Programming* koji je tiskan u klasičnom knjigotisku gdje je tiskarski slog slagan ručno. Do sljedećeg izdanja te knjige tehnologija u tisku se izmijenila i ručni tiskarski slog zamjenjuje fototipija te su svi originalni fontovi izgubljeni. Takva tehnologija također nije davala dovoljno kvalitetne rezultate koje bi zadovoljili Knuthove standarde. U to vrijeme digitalni tisak se nije smatrao kvalitetnim, no Knuth je saznao za jedan digitalni stroj koji zadovoljio njegove uvjete, a istovremeno bio potpuno digitalan. Tada se rodila ideja o stvaranju  $\TeX$ -a, slovoslagarskog programa koji je isprva trebao biti namijenjen samo njemu i njegovoj tajnici.[1]

Razvoj  $\TeX$ -a nije bio jednostavan i Knuthu je trebalo 10 godina da ga usavrši. Trenutno je aktivna verzija  $\TeX$  3.14159265 koja je simbolično preuzeta iz broja  $\pi$  te se za svako ažuriranje dodaje sljedeća znamenka iz tog broja. Knuthova ideja je da se nakon njegove smrti trenutna verzija  $\TeX$ -a prevede u  $\TeX\pi$  i da svi naknadni *bugovi* postanu karakteristika programa. [2]

### 2.2. Donald E. Knuth

Donald Erwin Knuth je računalni znanstvenik, matematičar i profesor emeritus na sveučilištu Stanford. Rođen je 10. siječnja 1938. godine u Milwaukeeu. Dobitnik je preko 100 nagrada, a 1974. godine dobio je nagradu Turing Award od Association for Computing Machinery (ACM), najprestižniju nagradu za računalnu znanost. Najpoznatiji je po seriji knjiga *The Art of Computer Programming* u kojoj je analizirao računalne algo-

ritme i njihovu strukturu. Knuth je i tvorca programskih sistema WEB i CWEB koji su primjer "pismenog programiranja". U njima koristi ugrađivanje izvornog koda unutar opisnog teksta, što je suprotno od uobičajene prakse. Razlika između ta dva jezika jest što je CWEB verzija WEB-a za C programski jezik. Knuth je u programskom sistemu WEB napisao svoje najpoznatije programe  $\text{\TeX}$  i METAFONT. [3]

Poznat je i po svom jedinstvenom humor, najpoznatiji je po čeku koji dodjeljuje onome tko pronađe grešku u njegovoj tipografiji ili knjigama. Ček iznosi 2.56 dolara jer je 256 penija zapravo jedan heksadecimalni dolar. Svake godine se za grešku pronađenu u kodu  $\text{\TeX}$ a ili Metafonta iznos čeka poduplavao, sve dok nije dostigao iznos od 327.68 dolara. U početku je Knuth izdavao čekove koje je sam dizajnirao sve do 2008. godine kada se zbog pojave prevara moralo ukinuti i uvesti čekovi banke San Serriffe. Izdao je više od 2000 čekova u vrijednosti od 20000 dolara, no rijetko koji ček je unovčen te ih vlasnici koriste kao trofej.[4]

Donald E. Knuth prikazan je na slici 1.



Slika 1. Donald Ervin Knuth 2005. godine,  
[https://en.wikipedia.org/wiki/Donald\\_Knuth#/media/File:  
KnuthAtOpenContentAlliance.jpg](https://en.wikipedia.org/wiki/Donald_Knuth#/media/File:KnuthAtOpenContentAlliance.jpg)

### 2.3. L<sup>A</sup>T<sub>E</sub>X i povijest L<sup>A</sup>T<sub>E</sub>X-a

L<sup>A</sup>T<sub>E</sub>X je besplatan software za obradu tekstualnog sadržaja i izradu dokumenata, a temelji se na jeziku T<sub>E</sub>X. Trenutno se koristi verzija L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> koja je prijelazna verzija iz L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>2.09 u novu, napredniju, L<sup>A</sup>T<sub>E</sub>X3. Za razliku od konvencionalnih programa za obradu teksta, kao što su Microsoft Word i Apple Pages, L<sup>A</sup>T<sub>E</sub>X ne prikazuje finalni izgled već se koristi *markup* oznakama, slično kao u HyperText Markup Language(HTML) i EXtensible Markup Language(XML) jezicima, za definiranje strukture dokumenta kao što su stil dokumenta (članak, knjiga,pismo,...), veličina i pored teksta, font, margine. U T<sub>E</sub>X svijetu bitno je razlikovati neke osnovne pojmove koji objašnjavaju funkciju programa. Važno je znati razliku između računalnog mehanizma (eng. *engine*) i formata. Računalni mehanizam je zapravo program, u ovom slučaju T<sub>E</sub>X, na čijem jeziku se temelje svi ostali formati kao što je L<sup>A</sup>T<sub>E</sub>X. Kod instalacije nekog od formata ujedno se instalira i osnovni T<sub>E</sub>X program.[5] Format je zapravo skup makro naredbi koje pojednostavljuju osnovne T<sub>E</sub>X naredbe i prilagođavaju ih tako da običnom korisniku budu razumljivije i logične.

Frank Mittelbach u intervju za Free Software Magazine kaže kako je L<sup>A</sup>T<sub>E</sub>X-ov koncept logičkog markup označavanja i njegov pristup pretvaranja logičkih oznaka u vizualnu reprezentaciju, utjecao ne samo na korisnike T<sub>E</sub>X-a već i na razvoj na HTML-a i raznih SGML/XML DTD-a.[6]

Zbog toga što se finalni izgled vidi tek na kraju generiranja dokumenta i što nudi tzv. *template* izgleda, L<sup>A</sup>T<sub>E</sub>X omogućava autoru da se fokusira isključivo na sadržaj. Upravo zbog svoje jednostavnosti ovaj program najviše je priznat u akademskom okruženju gdje se najčešće koristi za objavu znanstvenih istraživanja, članaka, radova, knjiga i sl. U L<sup>A</sup>T<sub>E</sub>X-u je znatno jednostavnije pisati matematičke formule, referencirati, citirati, pisati partiture, a uz sve to podržava složene jezične strukture koje se mogu naći u grčkom, arapskom i sličnim jezicima.[7]

Kako T<sub>E</sub>X inače proizvodi DVI (Device Independent file format) datoteke koje se danas rijetko koriste, za konvertiranje L<sup>A</sup>T<sub>E</sub>X datoteka u PDF koristi se tzv. L<sup>A</sup>T<sub>E</sub>X distribucije kao što su MikTeX, T<sub>E</sub>X live, L<sup>A</sup>T<sub>E</sub>X base, proTeXt itd. Postoje i online verzije kao što su Overleaf, ShareLaTeX, Papeeria itd.[8]

Ranih osamdesetih Leslie Lamport planira napisati knjigu Great American Concurrency Book. Koristeći  $\TeX$  morao je stvoriti novu makronaredbu za oblikovanje svoje knjige. Zaključio je da treba napisati makronaredbu koja će pojednostaviti  $\TeX$  i koju će moći koristiti i drugi. Tako se razvila ideja o  $\LaTeX$ -u te Lamport planira napisati priručnik za korištenje. Sumnjao je da se priručnik neće prodavati, no 1983. godine urednik njegove izdavačke kuće uvjerava ga da objavi priručnik koji će se kasnije prodati u tisućama primjeraka.[9]

Tada se za  $\LaTeX$  govorilo da je to „ $\TeX$  za masu“ i „Scribe<sup>1</sup> oslobođen nefleksibilne kontrole formatiranja“.[10]

U kolovozu 1989. godine, na sastanku TeX Users Group (TUG) Lamport predaje LaTeX Franku Mittelbachu koji zajedno s Chrisom Rowleyom i Rainer Schöpfom osniva  $\LaTeX$ 3 tim i 1994. izdaje trenutnu verziju  $\LaTeX$  2<sub>ε</sub>. Tim je nastavio raditi na budućoj  $\LaTeX$ 3 verziji.[11]

## 2.4. Ime $\TeX$ i $\LaTeX$

TeX u riječi LaTeX odnosi se na TeX program Donalda Knutha [12] Knuth u svojoj knjizi „The TeXbook“ kaže kako riječ TeX dolazi od grčke riječi umjetnost ili tehnologija te se izgovara kao teh. No, Leslie Lamport kaže da se izgovor najbolje određuje korištenjem imena pa tako nije neuobičajeno čuti i latek, lejtec i lejteks. [12]

Kod pisanja imena  $\LaTeX$  i  $\TeX$  zahtijeva se korištenje službenih logotipa, prikazanih na Slici 2 i 3, koji se u  $\LaTeX$ -u mogu pozvati jedonostavnim naredbama  $\backslash\LaTeX$  i  $\backslash\TeX$ . U slučaju kada to nije moguće, kao kod pisanja e-maila, imena se pišu „LaTeX“ i „TeX“.

Posebno je važno je da se  $\TeX$  ime piše pravilno kako bi se razlikovao od TEX (izgovara se teks) akronima za Text EXecutive processor. [13]

## 2.5. CTAN

CTAN ili The Comprehensive  $\TeX$  Archive Network je niz web stranica na kojima se mogu pronaći i skinuti materijali vezani uz  $\TeX$  akoder se preko nje mogu naći sve  $\TeX$  i

---

<sup>1</sup>Scribe je markap jezik i sustav za obradu teksta Briana Reida koji je uveo uporabu opisnog markap jezika



Slika 2.  $\LaTeX$  logo,

[https://bs.wikipedia.org/wiki/LaTeX#  
/media/Datoteka:LaTeX\\_logo.svg](https://bs.wikipedia.org/wiki/LaTeX#/media/Datoteka:LaTeX_logo.svg)



Slika 3.  $\TeX$  logo,

[https://bs.wikipedia.org/wiki/Datoteka:  
TeX\\_logo.svg](https://bs.wikipedia.org/wiki/Datoteka:TeX_logo.svg)

$\LaTeX$  distribucije. Prije CTAN-a postojalo je nekoliko ljudi koji su objavljivali  $\TeX$  materijale za širu publiku, ali nije bilo jedne određene stranice gdje su se svi materijali skupljali. Na konferenciji EuroTEX 1991. godine, nastala je ideja da se spoje odvojene zbirke. 1992. Rainer Schoepf i Joachim Schrod u Njemačkoj, Sebastian Rahtz u Velikoj Britaniji i George Greenwade u SAD-u su izgradili CTAN. Službeno je najavljen na EuroTEX konferenciji u Astonu, 1993. godine. Danas postoji preko 100 povezanih CTAN stranica na koje je objavljeno oko 5714 paketa od 2619 korisnika. Prosječno se objavi 100 paketa mjesečno.[14]

## 2.6. Razlika između klase i paketa

Kod korištenja LaTeX-a važno je razlikovati 2 pojma: class (.cls) i package(.sty).

LaTeX dolazi s nekim osnovnim predlošcima izgleda dokumenta. Kada postavljamo dokument, prije početka pisanja, pozivamo jedan takav predložak koji će pri izvoženju naš tekst oblikovati u grafički smislenu i lijepo uređenu cjelinu. Osnovni predlošci su article, report i letter. U LaTeX svijetu takvi predlošci se zovu klase (eng. *class*). U njima je definiran finalni izgled dokumenta; margine, font, veličina naslova i podnaslova, prored itd. Velik izbor predefiniranih izgleda dokumenata može se pronaći na CTAN-u. Ipak nekad ni tamo ne možemo naći nama potreban izgled, u tom slučaju možemo napisati svoju klasu. Prije početka pisanja klase važno je biti siguran trebamo li uistinu klasu ili nam je dovoljan package.

Pravilo je ako se nove naredbe koje želimo uvesti mogu koristiti u svakoj klasi dokumenta, onda trebamo paket. Ako su naše naredbe ipak samo preoblikovanje neke već postojeće klase, i ne mogu se koristiti u nekoj drugoj klasi, tada radimo novu „cls“ dato-

teku. Suprotno tome, paketi omogućuju korištenje nove naredbe neovisno o klasi. Takav primjer je paket „graphics.sty“ koji omogućuje ubacivanje slika u dokument. S obzirom na to da se može koristiti u svakoj klasi, logičnije je da je definiran kao .sty nego .cls datoteka.[15]

## 2.7. Prednosti i mane (La)TeX-a

Iako su WYSIWYG programi prevladali i skoro svatko se njima koristio barem jednom u životu, TeX i L<sup>A</sup>TeX ipak imaju svoje vjerne korisnike. Neke od prednosti koje oni pružaju su:

- svaki dokument izgleda profesionalno dizajniran,
- pisanje matematičkih formula je jednostavnije
- korisnik treba naučiti samo nekoliko naredbi koje definiraju strukturu dokumenta
- kompleksne strukture kao fusnote, sadržaj, referiranje, indeksi i sl. se jednostavno naprave
- TeX i L<sup>A</sup>TeX su potpuno besplatni i podržani su na svim operativnim sustavima
- Dokumenti napravljeni u tim programima obični su ASCII dokumenti i jednostavno se prenose s jedne na drugu platformu.

Ipak, ovi programi imaju i neke mane, npr. izrada potpuno novog izgleda dokumenta komplicirana je i složena, intuitivne stvari koje pišemo u WYSIWYG programima su u LaTeX-u malo „kompliciranije“. Tako se za novi red umjesto tipke „enter“ treba upisati "\\", ako želite da slovo iza točke ostane kurentno treba upisati ". \" , a za običan znak \$ ili vitičastu zagradu treba upisati "\\\$" i "\\{". Sve navedene mane logične su kada se zna pozadina rada L<sup>A</sup>TeX-a i jednom kada ih se savlada više ne predstavljaju problem.



## 2.8. Princip rada T<sub>E</sub>X-a

T<sub>E</sub>X<sup>2</sup> spada u klasu *softwarea* zvanih *kompajleri*. To su programi koji unose datoteku napisanu na izvornom jeziku i pretvaraju je u izlaznu datoteku napisanu na ciljnom jeziku. Točnije T<sub>E</sub>X je kompajler dokumenata. Za TeX mehanizme (*compiler*) ulazna datoteka zapisana je u T<sub>E</sub>X jeziku za pisanje i cilj je izlazna datoteka napisana na drugom “jeziku” kao što je Device Independent(DVI) ili Portable Document Format(PDF). Datoteka `.tex` je, naposljetku, jedan dugi niz znakova (uključujući znakove za prekid retka) koja sadrži tekst namijenjen za slaganje slogova sa znakovima: `\`, `{`, `$`, `[` te svim vrstama znakova koji se mogu pojaviti u naizgled gotovo beskonačnom rasponu kombinacija. Za obradu `.tex` datoteke T<sub>E</sub>X softver također mora pogledati svaki znak unutar vašeg unosa i dodijeliti mu dodijeliti značenje. Međutim, T<sub>E</sub>X je samo softver koji se bavi obradom teksta spremljenog kao niz brojeva (kodova) koji se nalaze u ulaznoj datoteci. T<sub>E</sub>X mora biti programiran s relevantnim podacima koji mu govore kako odrediti značenje znaka koji “gleda”, a zatim što treba učiniti s njim.[17]

T<sub>E</sub>X koristi kodove kategorija (*category codes*) kojih ima 16, u rasponu od 0 do 15. Svaki znak na koji T<sub>E</sub>X naiđe u `.tex` datoteci ima svoj unaprijed dodijeljeni kod kategorije. Unutar T<sub>E</sub>X softvera je svojevrsna “pregledna tablica” koja navodi popis kategorija koje su trenutno dodijeljene svakom znaku koji T<sub>E</sub>X može vidjeti unutar ulazne `.tex` datoteke. Kodovi kategorija zapravo pridodaju značenje svakom pojedinom znaku unutar *inputa* kojeg mora pregledati i određuje kako će ga T<sub>E</sub>X procesuirati. Preko kodova kategorije T<sub>E</sub>X zapravo razlikuje sadržaj kojeg treba posložiti od uputa/naredbi koje mora izvršiti.[17] U tablici 1 prikazani su svi kodovi i njihovo pridodano značenje.

Korištenje kodova je T<sub>E</sub>X-ov glavni mehanizam filtriranja dolaznog toka znakova dajući smisao unesenom tekstu i određujući: znakove koji sadrže tekst koji se snima; odvajanje matematičkog sadržaja; nizove znakova koji su imena naredbi koje treba izvršiti, te mnoge druge operacije slaganja sloga. Iako izgleda kao da je za svaki znak kod kategorije (značenje) uvijek isti i trajan to ipak nije tako. Kako je rečeno, T<sub>E</sub>X održava tablicu kodova koji su trenutno dodijeljeni nekom znaku, no to se može promijeniti primitivnom

---

<sup>2</sup>T<sub>E</sub>X je ujedno i računalni mehanizam i programski jezik, ovdje se to odnosi na mehanizam ili *engine*

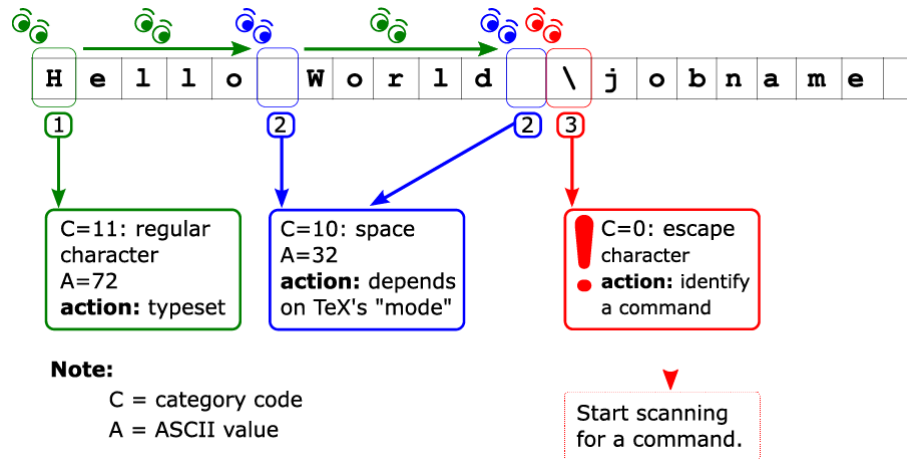
Tablica 1. 16 kodova kategorije (*category codes*) u  $\text{\TeX}$ -u,

[https://www.overleaf.com/learn/latex/How\\_TeX\\_macros\\_actually\\_work:\\_Part\\_1](https://www.overleaf.com/learn/latex/How_TeX_macros_actually_work:_Part_1)

Kod kategorije	Opis	Standardni $\text{\TeX}$ ili $\text{\LaTeX}$
0	Znak "za bijeg"	<code>\</code>
1	Početak grupiranja	<code>{</code>
2	Kraj grupiranja	<code>}</code>
3	Matematički mod	<code>\$</code>
4	Kartica poravnanja	<code>&amp;</code>
5	Kraj linije	ASCII code 13 ili "return"
6	Parametar	<code>#</code>
7	Matematički znak za potenciju: $y = x^2$	<code>^</code>
8	Matematički znak za indeks: $y = x_2$	<code>_</code>
9	Zanemareni znak	ASCII 0 <null>
10	Razmak	ASCII codes 32 (space) and 9 (tab)
11	Slovo	A...Z, a...z, (i tisuće Unicode znakova)
12	Drugo	0...9 plus „;?” i mnogi drugi
13	Aktivni znakovi	Posebni kodovi kategorije za stvaranje jednoznačnih makro naredbi kao ~
14	Znak za komentar — ignoriraj sve što slijedi ko kraja linije	<code>%</code>
15	Nevažeći znak, nije dopušten u u .tex ulaznom dokumentu	ASCII code 127 (DEL)

naredbom `\catcode`. Ta naredba omogućava potpuno mijenjanje kako  $\text{\TeX}$  tretira neki znak i nudi golem prostor za izmjenu sloga. Ova naredba je ono što omogućuje  $\text{\LaTeX}$ -u i ostalim paketima da zapravo budu funkcionalni.[17]

$\text{\TeX}$  svakom znaku pridodaje kod kategorije ali isto tako i heksadecimalni kod kojeg čita. Na slici 4 prikazano je što se događa kada naiđe na znakove.



Slika 4. Prikaz kako  $\text{\TeX}$  čita znakove,

[https://www.overleaf.com/learn/latex/How\\_TeX\\_macros\\_actually\\_work:\\_Part\\_2](https://www.overleaf.com/learn/latex/How_TeX_macros_actually_work:_Part_2)

Kada naiđe na znakove označene zelenom bojom  $\text{\TeX}$  će ih očitati kao znakove kategorije 11 (slovo) i proslijediti te znakove na slaganje sloga. Međutim, ne prosljeđuje ga samog već mu pridodaje brojevni par, kod znaka i kod kategorije koji zajedno čine token znaka (*character token*). Tek kada je izgrađen token, znak ulazi u proces slaganja.

Plava boja označava razmak (ASCII 32) s kategorijom 10. Kako  $\text{\TeX}$  zapravo procesuiru razmak ovisi o njegovom trenutnom "modu". Nekad će ga ignorirati, a u ovom slučaju će ga tretirati kao fleksibilni prostor kojeg može rastezati i smanjivati.

Crvena boja označava znak s kategorijom 0, "znak za bijeg". Svaki znak kategorije 0 daje  $\text{\TeX}$  uputu da se prebaci u poseban način čitanja koda jer on označava naredbu, a ne tekst. Nakon toga  $\text{\TeX}$  očitava kod kategorije znaka koji slijedi odmah iza znaka `\` iz razloga što  $\text{\TeX}$  prepoznaje 2 vrste naredbi:

- višeslovne naredbe ili tzv. kontrolne riječi. Svi sljedeći znakovi kategorije 11 se

smatraju dijelom naredbe, sve dok ne naiđe na znak neke druge kategorije. Kada naiđe na znak koji nije u kategoriji 11  $\TeX$  to shvaća kao kraj naredbe.

- jednoslovnu naredbu ili kontrolni simbol. Znak koji slijedi iza znaka `\nema` kategoriju 11.

Tokeni znakova stvaraju se prema formuli:

$$T = 256 \times C + A \quad (1)$$

gdje je  $C$ =kod kategorije, a  $A$ =Unicode vrijednost. Ova formula koristi se za  $\TeX$ , pdf- $\TeX$ , e- $\TeX$  i ostale *engine* osjetljive na Unicode znakove. Xe $\TeX$  i slični koriste drugačiju formulu:

$$T = 2^{21} \times C + Unicodevrijednost \quad (2)$$

Jednom dobiveni broj trajno ostaje vezan uz taj znak. Znakovi kategorije 0 se ne računaju,  $\TeX$  ih jednostavno prepoznaje kao prekidač za promjenu ponašanja.

Kada  $\TeX$  naiđe na naredbu on ju mora pronaći u svojoj pohrani i pravilno je provesti. Pronalazi ju tako da svaku naredbu, koja u početku bila samo niz cijelih brojeva, konvertira u jedan cijeli broj. Ta funkcija zove se *hash* te se nakon njene provedbe i broj naredbe sprema u varijablu zvanu *curcs* (current control sequence). Proces traženja daje  $\TeX$ -u još 2 broja, *curcmd* i *curchr* koja koristi za pronalaženje značenja naredbe. *Curcmd* ili trenutna naredba je cjelobrojna varijabla. Koristi se za pohranjivanje trenutne naredbene vrijednosti za naredbu koja se obrađuje ili pohranjuje trenutni kod kategorije znaka koji se obrađuje. *Curchr* ili trenutni znak (*current character*) je cjelobrojna varijabla koja pohranjuje ono što je  $\TeX$  zadnje pročitao a može biti: znak ako je posljednje očitana stvar bila znak u kom slučaju se pohranjuje trenutni kod znaka; ili naredba ako je posljednje očitana stvar bila naredba u kom slučaju pohranjuje naredbeni modifikator. Ako je  $\TeX$  slučajno pročitao znak, a ne naredbu, namjestit će da *curcs* ima vrijednost 0 da bi mogao zapamtiti da je posljednja pročitana stvar bila znak, a ne naredba. *Curtok* je trenutni token ili *current token*, cjelobrojna varijabla koja pohranjuje vrijednosti trenutnog tokena koji je ili naredbeni token ili token znaka. Za znakove najveća moguća vrijednost tokena u 8-bitnim računalnim mehanizmima se dobiva po formuli

$curtok_{max} = 256 \times 15 + 255 = 4049$ , a za naredbe  $curctok_{max} = 4049 + curcs$ . Prema tome ispada da sve što ima vrijednost veću od 4049 jest naredba, a manju je znak.

Četiri primitivne naredbe u TeX-u su `def`, `gdef`, `edef` i `xdef` koje imaju sličnu funkciju i sve definiraju makro naredbe. Da bi se mogle iskoristiti njihove individualne funkcije TeX dodjeljuje 2 vrijednosti svakoj naredbi i te vrijednosti daju naredbi značenje, njezinu ulogu. Te vrijednosti nalaze se duboko unutar TeX sistema i nisu dostupne korisniku. Svaka naredba, bila primitivna ili načinjena od strane korisnika, ima 2 vrijednosti: kod naredbe - generalnu klasifikaciju ili tip naredbe; modifikator naredbe - pomoćna informacija koja daje TeX-u dodatne informacije o naredbi. Zajedno naredbeni kod i modifikator naredbe daju jedinstvenu naredbu.[17]

Kod definiranja makro naredbi važno je razlikovati 2 pojma: argument i parametar. Ovdje se koristi primjer iz online članka *A six-part series: How do TeX macros actually work?* originalno dostupnog na [www.overleaf.com](http://www.overleaf.com). Ako se definira naredba `foo`

```
\def\foo#1#2{Ovo je #1, Ovo je #2} %do #9
```

konstrukcije (tokeni) `#1` i `#2` zovu se parametri koji su "rezervna mjesta" za stvarne podatke koji će se koristiti, npr.

```
\foo{alpha}{beta}
```

gdje su *alpha* i *beta* argumenti odnosno stvarne vrijednosti koje se umeću na mjesto `#1` i `#2`.

Svaka makro definicija sastoji se od 4 dijela:

```
<TeX makro primitiv><makro ime><parametar tekst>{<zamjenski tekst>}
```

gdje je

TeX makro primitiv jedan od primitiva: `\def`, `\edef`, `\gdef` or `\xdef`,

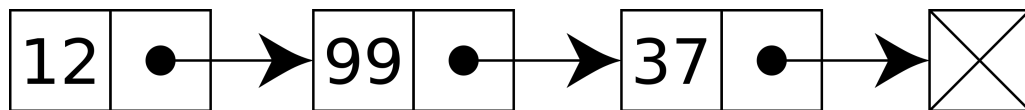
makro ime: ime makro naredbe, npr. `\foo`;

parametar teksta: može biti izostavljeno, to je serija tokena koji se pojavljuju prije <zamjenskog teksta>, npr. `#1,#2,...#9`. U suštini parametar teksta pruža predložak to-

kena koji pomaže  $\TeX$ -u da shvati koji token korisnik želi koristiti kao makro argument;

zamjenski tekst: ovo je tijelo makro naredbe, to je niz tokena unutar kojih se dodaju argumenti kada je makro naredba procesuirana. Argumenti se dodaju u pozicije označene makro parametrima u originalnoj definiciji.[17]

Za pohranu popisa tokena (cjelobrojnih vrijednosti)  $\TeX$  koristi strukturu podataka nazvanu povezani popis (eng. *linked list*), koji u  $\TeX$ -ovom slučaju sadrži popis takozvanih čvorova (eng. *nods*). Čvor je mali paket memorije računala koji se može koristiti za spremanje zbirke podataka. Da bi se makro pohranio, ti su čvorovi povezani kao lanac, gdje svaki čvor (veza u lancu) može pohraniti nekoliko dijelova informacija - uključujući vrijednost tokena i memorijsku adresu sljedećeg čvora na popisu.[17] Primjer kako povezani popis prikazan je na slici 5.



Slika 5. Povezani popis čiji čvorovi sadrže dvije značajke; cijeli broj i vezu na sljedeći čvor, [https://en.wikipedia.org/wiki/Linked\\_list#/media/File:Singly-linked-list.svg](https://en.wikipedia.org/wiki/Linked_list#/media/File:Singly-linked-list.svg)

Kada  $\TeX$  definira makronaredbu, dijelovi  $\langle$ parametar teksta $\rangle$   $\langle$ zamjenski tekst $\rangle$  pretvaraju se u jedan dugi kontinuirani popis oznaka - ukupni broj tokena na tom popisu ovisi o složenosti makronaredbe. Odjeljak  $\langle$ parametar teksta $\rangle$  ima specifičnu svrhu da djeluje kao "predložak tokena" ili "nacrt" koji  $\TeX$  koristi za odabir tokena koji formiraju argumente (vrijednosti) za upotrebu sa stvarnom makronaredbom: tj. tokeni koji se unose u  $\langle$ zamjenski tekst $\rangle$ . [17]

## 2.9. Struktura $\LaTeX$ -a

Kod pisanja  $\LaTeX$  dokumenta vrlo je važno pratiti osnovnu strukturu bez koje se dokument ne može izraditi.  $\LaTeX$  ima dva osnovna strukturalna dijela: preambulu i tijelo. Preambla ili zaglavlje  $\LaTeX$  jest prvi dio dokumenta u kojem se pozivaju sve važne funkcije za normalan rad programa. Počinje s pozivanjem naredbe:

```
\documentclass[argumenti]{naziv}
```

Bez te naredbe  $\LaTeX$  ne može izraditi dokument jer nema predložak po kojem bi oblikovao tekst. Obično se tu pozivaju neke od klasa kao što su `article`, `letter` ili `book`. U preambli se pozivaju i svi stilovi (paketi) koji se koriste unutar dokumenta. Obično je to naredba `\usepackage{naziv}`.

Drugi dio  $\LaTeX$  ulazne datoteke jest tijelo (eng. *body*) koje je uokvireno naredbom:

```
\begin{document}... \end{document}
```

U prostor između `begin` i `end` upisujemo sav sadržaj koji želimo da se oblikuje, odnosno tekst.

Sve naredbe se pozivaju uz pomoć znaka `\`. Naredbe mogu biti same za sebe (npr. `\newpage`, `\section`, `\includegraphics`) ili unutar okruženja. Okruženje najčešće počinje i završava s riječima *begin* i *end*, npr. `\begin{figure}\end{figure}` Unutar vitičastih zagrada `{ }` upisuju se nazivi naredbi ili njihova vrijednost. U uglate zagrade `[]` upisuju se opcije koje naredba omogućava, npr. veličinu teksta.

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

### 3. EKSPERIMENTALNI DIO

U današnje vrijeme dizajn igra važnu ulogu u prezentaciji rada, proizvoda ili samoga sebe. Osim dobre teme i sadržaja važno je dokument dobro oblikovati da čitatelju čitanje bude što lakše i ugodnije. Autori znanstvenih radova, knjiga ili članaka često nemaju vremena za estetiku pa im u pripomoć dolaze programi kao LaTeX - sustav pripreme i formatiranja dokumenata za tisak. Za razliku od „What you see is what you get“ programa kao što su Adobe InDesign, Microsoft Word, Apple Pages i sl., LaTeX se fokusira isključivo na sadržaj, a njegov izgled se uređuje markup označavanjem u jeziku TeX. Tako se autor može posvetiti samom sadržaju, a LaTeX će za njega, prema unaprijed pripremljenom predlošku, oblikovati izgled teksta.

Cilj ovog diplomskog rada jest izraditi  $\LaTeX$  klasu koju će budući studenti Grafičkog fakulteta moći koristiti za oblikovanje svojih završnih i diplomskih radova. Utvrdit će se i razina jednostavnosti korištenja programa u odnosu na „What you see is what you get“ program, npr. Microsoft Word. Prva hipoteza ovog diplomskog rada je da LaTeX program omogućava korisniku da se usredotoči na sam sadržaj kojeg piše ne razmišljajući o grafičkom oblikovanju te time doprinosi samoj kvaliteti sadržaja i krajnjeg proizvoda. Druga hipoteza je da  $\LaTeX$  pruža jednako kvalitetne mogućnosti kod oblikovanja teksta kao i standardni programi za uređivanje teksta. Treća hipoteza je da je  $\LaTeX$  jednostavan za korištenje, odnosno da ga može koristiti prosječna osoba bez predznanja u programiranju.

Za početak je bilo važno savladati osnovne naredbe i funkcionalnosti  $\LaTeX$ -a jer nije postojala podloga ni početno znanje do početka pisanja ovog rada. Za vježbu je uzeto oblikovanje Lorem Ipsum teksta u *article* klasi gdje su se mijenjale karakteristike teksta i eksperimentiralo se s naredbama. Može se reći da je znanje o  $\LaTeX$ -u, u trenutku početka pisanja klase, bilo na razini minimuma te se učilo „u hod“ uz pomoć mentora.

Za ovaj diplomski rad odabran je online program Overleaf iz razloga što omogućuje suradnju više autora. U ovom slučaju to je važno kako bi mentor mogao u svakom trenutku



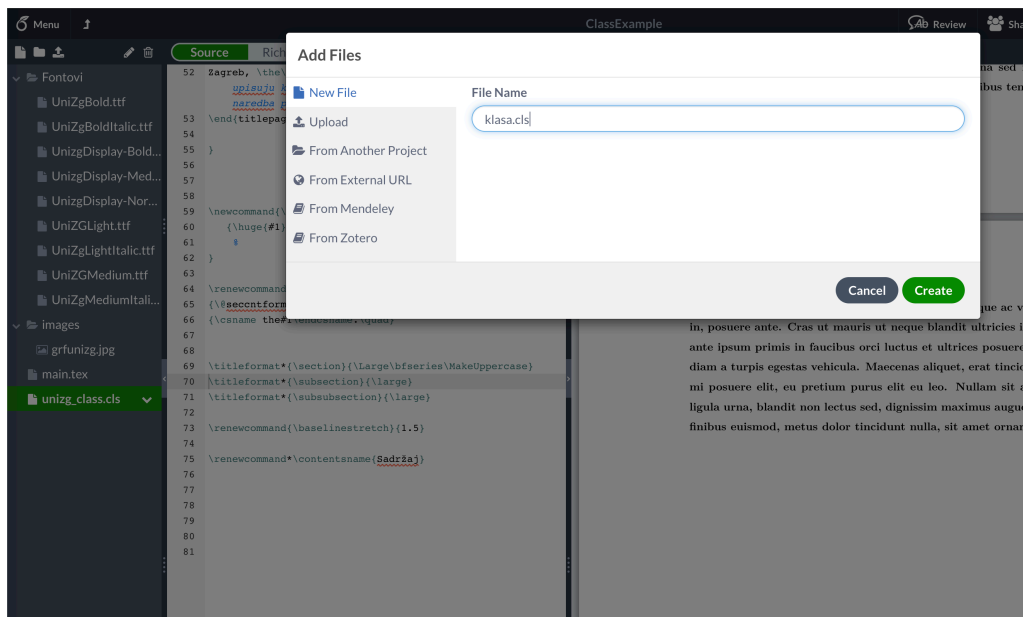
pregledati rad na jednostavan način, bez slanja dokumenata s kodom preko e-maila ili nekih online servisa za dijeljenje dokumenata. Također nudi mogućnost komentiranja bez  $\LaTeX$  inputa, pregled povijesti, *chat*, jednostavan upload na internet, grafički lijepo uređeno sučelje i *built-in* pregled generiranog PDF dokumenta te grešaka u kodu. Olakšavajuća mogućnost je i biranje *compilera* koji je u ovom slučaju XeLaTeX. XeLaTeX nije značajno drugačiji od  $\LaTeX$ -a no ima važnu značajku podržavanja UTF-8 znakova, odnosno hrvatskih dijakritičkih znakova (ć,č,đ,š,ž) bez uvođenja novih paketa. Također je važna značajka jednostavnog korištenja True Type fontova (.ttf) i Open Type Fontova (.otf) koje pdfLaTeX odnosno  $\LaTeX$  ne podržava i nije ih moguće koristiti. Iz razloga što se u ovom radu koristi se UniZg font ta značajka nije bila zanemariva.

### 3.1. Postavljanje klase

Dokument klase (eng. *class* s ekstenzijom .cls) nije ništa drugačiji od običnog  $\LaTeX$  dokumenta(.tex), ne zahtijeva nikakve nove, drugačije programe i aplikacije već se piše u istom programu kao dokument kojeg želimo oblikovati. Glavna razlika je njegova ekstenzija .cls te ono što se u njega upisuje. U taj dokument upisuje se sve ono što se inače upisuje u preambli .tex dokumenta objašnjenom u poglavlju 2.9 na stranici 15. *Class* dokument nije potrebno pisati svaki puta kada se radi novi dokument u  $\LaTeX$  iz razloga što se velika većina korekcija ili estetskih zahvata može raditi u većini klasa, bila ona osnovna ili neka nova. Klasu je najbolje izraditi kada je sigurno da rješenje već ne postoji na CTAN-u, a koristit će se često, takav dokument bi bio npr. predložak za oblikovanje izvješća neke firme. Klasu je ponešto teže izraditi nego običan  $\LaTeX$  dokument.

Klasa se može graditi već na postojećoj klasi ili se može iznova "programirati" u  $\TeX$  jeziku. Programiranje iznova je mukotrpano i zahtijeva dobro poznavanje kodiranja u  $\TeX$ u i  $\LaTeX$ -u. U ovom radu odabrano je raditi klasu na već postojećoj, osnovnoj *article* klasi iz razloga što je ta klasa dovoljno slična željenom izgledu UniZG klase te nisu potrebne dubinske, teške korekture.

Stvaranje *class* dokumenta u Overleaf-u je jednostavno, klikom na gumb "New file" otvara se izbornik prikazan na slici 6 u koji se mora umjesto ekstenzije .tex upisati .cls te upisati željeno ime klase. Klikom na "save" stvara se novi, potpuni prazan dokument



Slika 6. "New file" izbornik

vidljiv u izborniku s lijeve strane slike 6.

Pisanje klase započinje se naredbom `\NeedsTeXFormat{LaTeX2e}` koja upućuje na  $\text{\LaTeX}$  verziju koja je potrebna za izvršavanje klase. Zatim se piše

`\ProvidesClass{Imeklase}[datumizrade]`

tako da se u glavnom (*main*)  $\text{\LaTeX}$  dokumentu ta klasa može pozvati koristeći njeno ime.

$\text{\LaTeX}$  je općenito osjetljiv na verzalna i kurentna slova u imenima pa je važno točno upisati naredbu, ali i ime klase, inače se naredba neće izvršiti. Sljedeći korak je pozvati klasu na kojoj se radi nadogradnja ili ju se izmjenjuje. To se radi naredbom

`\LoadClass[imeklase]`. Nakon te dvije naredbe mogu započeti izmjene.

### 3.2. Izrada naslovnica

Prema uputama Grafičkog fakulteta koje se mogu pronaći na linku <https://www.grf.unizg.hr/obraci/> izrađene su dvije naslovne strane, odnosno određen je izgled teksta na tvrdom uvezu (koricama) i na naslovnoj strani rada. Prema uputama stranice diplomskog rada moraju biti A4 formata (210x297 mm), lijeva i donja margina moraju biti 3,5 cm, gornja i desna margina 2,5 cm. Tekst mora biti veličine 12 pt za tijelo dokumenta, naslovi moraju biti ispisani kurentnim, bold slovima, a podnaslovi verzalnim slovima. Tekst mora biti lijevo ili obostrano isključen i prored namješten na 1,5. Litera-

tura mora biti obilježena rednim brojem po redosljedu citiranja, a u tekstu označena uglatim zagradama.

### 3.2.1. Dimenzije i margine

Za početak su definirane dimenzije i margine dokumenta. Prema uputama to je A4 papir (210x297mm) s marginama: lijeva 35mm, desna 25 mm, gornja 25 mm, donja 35 mm. Kako bi svi diplomski radovi bili ujednačeni ovaj parametar je namjerno upisan u klasu, a ne ostavljen na izbor korisniku.  $\LaTeX$  ne dopušta mijenjanje margina kod pozivanja klase već se za to mora koristiti jedan od mnogih paketa, ovdje je korišten *Geometry*. Taj paket odabran je iz razloga što je to jedan od paketa koji se najčešće koriste te je dovoljan za ovu svrhu. Iako se upisuje u .cls datoteku poziva se kako bi se i inače pozivao, naredbom `\usepackage{geometry}` ili `\RequirePackage{geometry}`. Razlika između tih dviju naredbi jest da se `\RequirePackage` uglavnom koristi kod pisanja klase i paketa jer se može upisivati prije `\documentclass{}` naredbe, dok se `\usepackage{}` ne može. U suštini te dvije naredbe rade istu stvar, ubacuju paket u dokument.

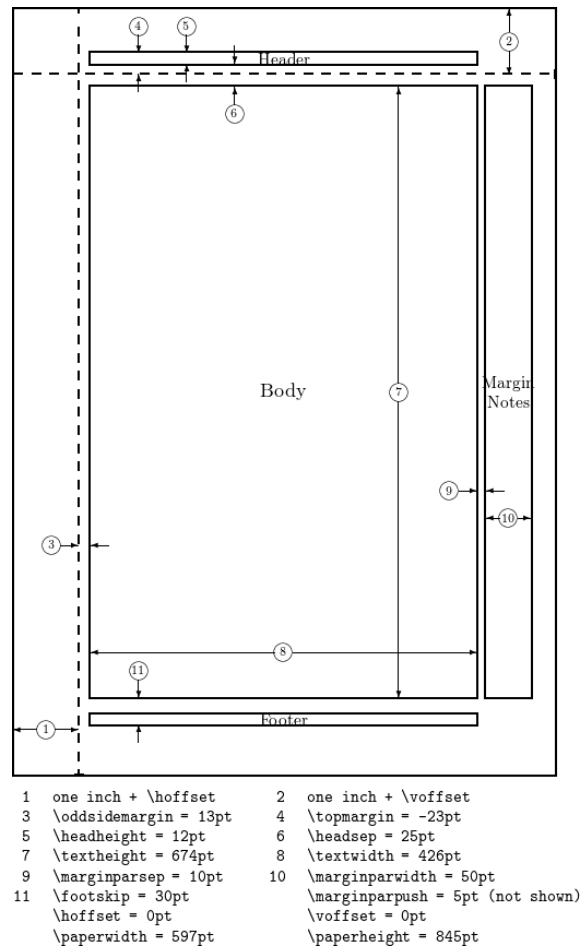
Da bi se točno specificirale veličine margina potrebno je bilo dati uputu koja se piše na ovaj način:

```
\usepackage{geometry}
\geometry{
a4paper,
left=35mm,
right=25mm,
top=25mm,
bottom=35mm
}
```

Ovo nije jedini način pisanja parametara, svi ostali načini navedeni su u uputama za korištenje koje dolaze s makro paketom *Geometry*, dostupan na <https://ctan.org/pkg/geometry>. Svi parametri moraju biti odvojeni zarezima te napisani bez razmaka.

Kako se diplomski radovi tiskaju jednostrano margine su definirane kao lijeva i desna, u

slučaju kada se tiskaju obostrano treba ih definirati kao unutarnju(*inner*) i vanjsku(*outer*). Na slici 7 prikazano je kako su u  $\text{\LaTeX}$ -u zadane margine i ostali parametri oblika stranice s početnim vrijednostima. Geometry paket može upravljati većinom parametara. Kod ovog diplomskog nije bilo važno da li su podnožje i zaglavlje uključeni u *body* stranice pa se zadana vrijednost nije mijenjala.



Slika 7. Zadane vrijednosti stranice u  $\text{\LaTeX}$ -u, <https://cdn.sharelatex.com/learn-scripts/images/f/fc/Layout-dimensions.png>

### 3.2.2. Ubacivanje UniZg fonta

Standardni  $\text{\LaTeX}$  kompajler ne podržava ubacivanje fontova već je početni font tzv. Roman, te se još može birati između San Serif i Typewriter fontova naredbama `\sffamily` i `\ttfamily`.

Iz toga razloga u ovom diplomskom koristi se XeLaTeX kompajler koji podržava True Type i Open Type fontove. Svakako se i u ovom slučaju mora pozvati makro naredba `\usepackage{fontspec}` za specificiranje novog fonta koji će se koristiti kroz cijeli dokument. Potrebno je specificirati sve verzije fonta koje bi se mogle koristiti, odnosno *bold* i *italic*. Kod korištenja Overleaf-a važno je font uploadati na stranicu preko gumba za upload. Kada se font uploada na stranicu bit će prikazan u izborniku gdje se nalaze i .cls i .tex datoteke. Važno je znati da sve klase koje se pišu sa svrhom dodavanja u CTAN repozitorij moraju biti neovisne o kompajleru. Zbog toga se paketi koji ovise o kompajleru, kao `fontspec` ne bi trebali dodavati u .cls datoteku već bi se trebali ostaviti korisniku na izbor iz jednostavnog razloga što ne koriste svi  $\text{\LaTeX}$  korisnici isti kompajler, u ovom slučaju XeLaTeX.

Iako je prvotna ideja bila pomoći budućim hrvatskim  $\text{\LaTeX}$  korisnicima i ubaciti UniZG font unutar .cls dokumenta, ipak je odlučeno premjestiti tu naredbu u preambulu .tex dokumenta kako potencijalne internacionalne kolege ne bi imali neželjene probleme s klasom.

Kod koji je korišten za ubacivanje UniZg fonta ide ovako:

```
\usepackage{fontspec}
\setmainfont{UniZGLight.ttf}[
  BoldFont=UniZgBold.ttf,
  ItalicFont=UniZgLightItalic.ttf]
```

U vitičaste zagrade upisuje se ime onog fonta koji želimo da bude za tijelo teksta, a u uglate zagrade specificiramo verzije za naglašavanje, podebljani i kurzivni font.

### 3.2.3. Definiranje tokena i novih naredbi

TeX je program koji djeluje logički te uvijek mora postojati jasna struktura kako bi on mogao posložiti dokument da izgleda dobro. Ponekad on ne pruža točno ono što je potrebno pa se za to koriste novi paketi i naredbe. No kada je slučaj da se te naredbe pojavljuje mnogo puta kroz dokument jednostavnije je napisati novu naredbu.

Za takvo što moguće je koristiti TeX naredbe kao što su `newcommand` i `renewcommand`. Za definiranje nove naredbe koristi se

```
\newcommand{naziv}[broj][osnovna postavka]{definicija}
```

gdje je *naziv* zapravo ime nove naredbe koje u sebi mora sadržavati `\` a ne smije sadržavati `\end`. *Broj* je broj parametara koja nova naredba ima i on može biti od 0 do 9 te nije obavezan u slučaju kad je 0. *Osnovna postavka* e zadana vrijednost za prvi parametar. To je ono što prvi parametar čini neobveznim, a ako nije definirano drugačije kod pozivanja naredbe, koristit će se ova zadana vrijednost. *Definicija* je ono što nova naredba zapravo radi. Istu sintaksu koristi i `\renewcommand` koja služi za preoblikovanje postojećih TeX naredbi.

U ovoj *unizg* klasi ta naredba koristila se za definiranje izgleda naslovne strane:

```
\renewcommand{\titlepage}{  
  \centering  
  \thispagestyle{empty}  
  \fontsize{24}{25}  
}
```

Naredba `\renewcommand` korištena je za preoblikovanje parametara naredbe `\titlepage`. U ovom primjeru *naziv* i dalje ostaje `titlepage`, *broj* parametara nije zadan, a u *definiciji* je određeno da će sada ta naredba generirati naslovnu stranicu :

- s centriranim tekstom
- bez numeriranja (*thispagestyle(empty)*)
- s veličinom pisma 24 pt te *baselineskip* 25 pt

Još jedan primjer korištenja iste naredbe jest za podešavanje proreda koji prema uputama za izradu diplomskog rada mora biti 1.5 pa kod izgleda ovako:

```
\renewcommand{\baselinestretch}{1.5}
```

gdje je upisana samo definicija, odnosno vrijednost naredbe. Iz toga je jasno da su parametri *broj* i *default* opcionalni.

Još jedan način definiranja novih naredbi su tzv. tokeni. Kada T<sub>E</sub>X čita redak teksta iz datoteke ili redak teksta koji ste unijeli izravno u terminal, pretvara taj tekst u popis "tokena". Token je ili jedan znak s pridruženim kodom kategorije ili je kontrolna sekvenca. Za stvaranje novog tokena poziva se naredba `\newtoks\imenaredbe`. S obzirom na to da je T<sub>E</sub>X primitivna sintaksa je malo drugačija od uobičajenih naredbi. S ovom naredbom T<sub>E</sub>X samo stvara prostor gdje će se novi token spremiti. U ovom radu tokeni su stvoreni u svrhu lakšeg oblikovanja naslovne strane tako da će korisnik u premablu svog dokumenta upisati vrijednosti tokena, odnosno svoje podatke. Rezultat će biti do kraja oblikovana naslovna strana i korice.

### 3.2.4. Oblikovanje naslovne strane i prve stranice

Nakon definiranja osnovnih postavki naslovnica te tokena definirao se izgled prve naslovne strane, odnosno korica. Kod u klasi izgleda ovako:

```
%-----definiranje kako title page izgleda---  
\renewcommand{\titlepage}{  
  \centering  
  \thispagestyle{empty}  
  \fontsize{24}{36}  
}  
  
%--oblikovanja veličine naslova-----  
\newcommand{\ime}[1]{  
  {\Large{#1}}
```

```

}

\newcommand{\uni}[1]{
  {\huge{#1}}
}

%----def prve naslovne stranice
\newcommand{\prvanaslovna}
{
  \begin{titlepage}
\uni{\the\sveuciliste}
\\
\uni{\the\fakultet}
\\
\vspace{4cm}
\ime{\the\imestudenta}
\\
\vspace{4cm}
\uni{\the\naslovrada}
\\
\vspace{36pt}
\ime{\the\rad}
\vfill
{\the\grad}, \the\year.
\end{titlepage}
}

```

Kao što je gore već navedeno naredba `\renewcommand{titlepage}` za funkciju ima centriranje teksta, određivanje veličine teksta na naslovnim stranama te uklanjanje numeracije. Nakon toga se stvorila nova naredba "ime" koja za definiciju ima naredbu `Large` odnosno veličinu pisma 17 pt ( to je samo u slučaju kada je za *article* klasu de-



finirano da je veličina pisma 12 pt, s promjenom osnovne veličine i *Large* se mijenja) te naredba "uni" s veličinom pisma 25 pt ili *huge*. Kasnije će te naredbe biti dodane prethodno definiranim tokenima *imestudenta,sveuciliste,naslov rada* itd.

Slijedi definiranje nove naredbe "prvanaslovna" koja će u finalnom dokumentu zapravo biti vanjski omot odnosno korice. Stvorila se naredba uz pomoć `newcommand` unutar koje su dane naredbe:

1. stvori *titlepage*
2. upiši naziv sveučilišta i pridodaj mu karakteristike naredbe "uni" te započni novi red
3. ubaci vertikalni razmak od 4 cm
4. upiši ime studenta i pridodaju mu karakteristike naredbe "ime"
5. ubaci vertikalni razmak od 4 cm
6. upiši naslov rada i pridodaj mu karakteristike naredbe "uni"
7. ubaci vertikalni razmak od 36 pt
8. upiši vrstu rada i pridodaju mu karakteristike naredbe "ime"
9. ubaci vertikalni razmak do kraja stranice
10. upiši ime grada i trenutnu godinu
11. završi *titlepage*

Slično se događa i kod definiranja druge naslovne strane odnosno,prve unutarnje strane. Glavna razlika jest u ubacivanju logotipa sveučilišta i fakulteta.

```

%----definiranje druge naslovne stranice----
\newcommand{\druganaslovna}{
  \begin{titlepage}

{\the\logo}
\\
\vspace{2.5cm}
\ime{\the\imestudenta}
\\
\vspace{3cm}
\uni{\the\naslovrada}
\\
\vspace{36pt}
\ime{\the\rad}
\vfill
Mentor:
\hfill
Student:
\\
{\the\mentor}
\hfill
{\the\imestudenta}
\vspace{24pt}
\\
{\the\grad}, \the\year.

\end{titlepage}
}

```

Nezgodan detalj na ovoj stranici jest pisanje riječi "mentor" i "student" u istom retku, ali s drugačijim poravnanjem. U WYSIWYG programu Microsoft Word takvo što se dobiva

tabulatorom (tipka *tab*) no u  $\LaTeX$ -u takav način unosa nije moguć pa je najjednostavniji način bio umetnuti `\hfill` (eng. *horizontal fill* - vodoravna ispuna) koji funkcionira da po x-osi ubacuje prazan prostor između dviju riječi tako da svaku riječ gurne do margina, u slučaju kada je više riječi on će ih horizontalno poravnati da razmak među riječima bude jednak.

U .tex datoteci, u preambli, su tokenima nadodane ove vrijednosti:

```
\imestudenta{Matea Milin}
\sveuciliste{Sveučilište u Zagrebu}
\fakultet{Grafički fakultet}
\naslovrada{\LaTeX{} \MakeUppercase{program za oblikovanje dokumenata i
izrada \LaTeX{} predloška za diplomski rad}}
\rad{Diplomski rad}
\mentor{Doc.dr.sc.Ante Poljičak}
\grad{Zagreb}
\logo{\includegraphics [width=8cm] {Slike/logo/grfunizg.jpg}}
```

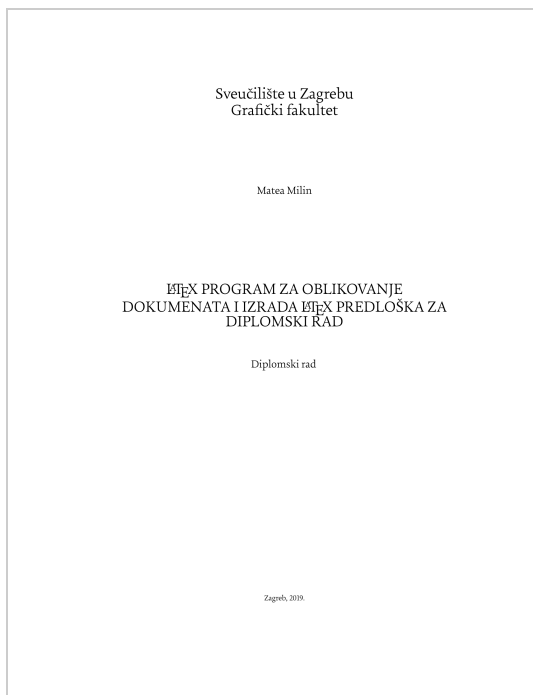
Najbitnije je bilo točno adresirati gdje se nalaze slike koje želimo ubaciti za logo fakulteta.

U ovom slučaju logo fakulteta se nalazi u mapi "logo" unutar mape više razine "Slike".

Zatim su pozvane naredbe:

```
\prvanaslovna
\druganaslovna
```

Takav kod stvorio je izgled prikazan na slikama 8 i 9.



Slika 8. Korice



Slika 9. Naslovna strana

### 3.3. Oblikovanje teksta dokumenta

Sljedeća važna stvar bila je uskladiti oblikovanje teksta koje proizvodi još uvijek aktivna *article* klasa s uputama grafičkog fakulteta.  $\text{T}_{\text{E}}\text{X}$  i  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}^3$  u sebi podržavaju samo ASCII znakove što stvara velike probleme za UTF-8 jezike s posebnim znakovima kao što je hrvatski. Da bi se tekst na hrvatskom prikazivao točno važno je učitati paket `\usepackage[croatian]{babel}`. Taj paket ujedno omogućuje korištenje dijakritičkih znakova, provjeru gramatike na hrvatskom jeziku, pravilno rastavljanje na slogove te automatski prevodi nazive kao što su *Figure* u "slika", *bibliography* u "Literatura" i sl. Moguće je učitati više jezika u slučaju da je dokument višejezičan te ih po potrebi izmjenjivati unutar dokumenta.

Sljedeći vrlo važan paket jest

`\usepackage{graphicx}`, poboljšanje prvotnog `graphicx` paketa, a koji omogućuje ume-tanje i upravljanje slikama u dokumentu. Bez jednog od tih paketa slike se ne mogu uči-tati.

<sup>3</sup> *Enginei* kao što su XeLaTeX ili LuaTeX sami po sebi podržavaju UTF-8 znakove

U datoteku `unizg_klasa.cls` je taj paket učitani preko naredbe

`\RequirePackage{graphicx}` no ono što je ipak ostalo na korisniku jest da pokaže put  $\TeX$ -u kako doći do slika. To se radi na način da se u preambli dokumenta upiše naredba `\graphicspath{NazivMape}`.

Kada je umetanje slika omogućeno zgodno je odmah nadodati i paket `caption` i `subcaption` koji nude mnogo načina za prilagodbu opisa u *float* okruženjima kao što su slika i tablica te se može uskladiti s mnogim drugim paketima. Bez uređivanja naslovi i podnaslovi u

*article* klasi izgledaju kao na slici 10. Takav format vrlo je sličan onome koji je definiran u uputama no ipak neki detalji nedostaju. Na podnaslovima svih razina nedostaju točke iza broja pa se tom problemu doskočilo na ovaj način:

```
\renewcommand
{\@secntformat}[1]
{\csname the#1\endcsname.\enskip} %dodavanje točke na section i
%malog razmaka(enskip)
```

Naredba `section` je preuređena prema uputama ranije učitanih paketa `titlesec` za uređivanje naslova.

S tim istim paketom preuredile su se i veličine pisma u svakom od podnaslova te se u `\section` još dodao i bold format te naredba da se uvijek piše verzalnim slovima.

Kako za neka poglavlja nije poželjno da imaju redni broj, kao npr. Sažetak i Sadržaj, kod upisivanja koda mora se dodati znak `*` koji daje uputu da se na njima izbriše redni broj, npr. `\section{imesekcije}`. U `unizg_klasa.cls` to izgleda ovako:

```
\titleformat*{\section}{\LARGE\bfseries\MakeUppercase{}}
\titleformat*{\subsection}{\large}
\titleformat*{\subsubsection}{\large}
```

Zadana je i veličina proreda, u *article* klasi to je 1.2, a za `unizg_klasa` potreban je prored 1.5. Korištena je već dobro poznata naredba `\renewcommand{\baselinestretch}{1.5}`.

Uvlake su također prethodno definirane. Naredba za njihovo uklanjanje ide ovako:

# Primjer L<sup>A</sup>T<sub>E</sub>X dokumenta

Matea

June 2019

## 1 Uvod

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

### 1.1 Podnaslov

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Slika 10. L<sup>A</sup>T<sub>E</sub>X *default* izgled naslova u *article* klasi

```
\setlength\parindent{0pt}%naredba za brisanje uvlaka
```

Kako je vidljivo na slici 7 numeracija je zadana da bude na sredini stranice. Za unizg klasu važno je bilo definirati numeraciju da bude na desnoj strani. Za to je korišten `\usepackage{fancyhdr}`. Taj paket dolazi s "ukrasima" koji nisu potrebni za unizg klasu pa je bilo potrebno redefinirati naredbu na ovaj način:

```
%----- definiranje numeriranja na desnu stranu-----  
\RequirePackage{fancyhdr}  
\pagestyle{fancy}  
\renewcommand\headrulewidth{0pt} % brisanje linije u headeru  
\fancyhead{} %prazan header  
\fancyfoot{} % prazan footer  
\fancyfoot[R]{\thepage} % numeracija na desnoj strani
```

Za poglavlja Sažetak i Sadržaj izbjegava se numeracija pa se u tom slučaju kod pozivanja naredbe za početak poglavlja treba dodati naredba `\thispagestyle{empty}`. Na onoj stranici na kojoj želimo da počne numeriranje upisuje se `\setcounter{page}{1}` gdje *page* označava trenutnu stranicu, a broj 1 onaj broj s kojim želimo da počne numeracija. U UniZG klasu sa svrhom oblikovanja ovog diplomskog rada dodatno su još ubačeni paketi *upgreek*, *hyperref*.

Originalno su bili ubačeni i paketi za prikaz programerskih i  $\LaTeX$  kodova no kako takvi paketi funkcioniraju uz pomoć Python modula *Pygments* odlučeno ih je izbaciti iz klase i ostaviti korisniku na izbor. Razlog tome jest što ukoliko korisnik nema instaliran *Pygments* na svoje računalo paketi *minted* i *tcolorbox* stvaraju greške i  $\LaTeX$  ne može procesuirati dokument.

Paket *tcolorbox* omogućuje dodavanje raznih obojenih okruženja, okvire s naslovom i sl. Primjer korištenja *tcolorbox*-a vidi se na slikama 8, 9 i 10 gdje je na originalnu sliku dodan sivi okvir kako bi se ista mogla razlikovati od pozadine. Korišten je i za istovremeni prikaz koda i *outputa* u dodatku A.2.1 ovog rada. Taj paket zahtijeva paket *minted* kao svoj motor za prikaz koda.

Paket *minted* služi za prikazivanje sintaksi i kodova nekih od programskih jezika kao što

su Python, JavaScript, MatLab ali i  $\LaTeX$ . Kroz ovaj rad korišten je skoro za svaki prikaz  $\LaTeX$  koda. Iznimka je prikaz koda korištenjem `\verb` sintakse za kraće prikaze koda i specijalne znakove.

Paket *upgreek* ubačen je za lakše umetanje grčkih znakova  $\alpha$ ,  $\beta$ ,  $\delta$  itd.

Paket *hyperref* se inače koristi za rukovanje unakrsnim referenciranjem u  $\LaTeX$  za izradu *hypertext* veza u dokumentu. U ovom radu je korištena samo mala popratna mogućnost tog paketa - usklađivanje fonta za prikaz URL veza sa zadanim fontom, UniZg fontom. To je postignuto naredbom:

```
\RequirePackage{hyperref}%prikazuje url kao link  
\urlstyle{same}%naredba da koristi isti font kao u dokumentu
```

### 3.4. Ubacivanje literature

Kako je rečeno na početku ovog rada,  $\LaTeX$  je iznimno popularan u akademskoj zajednici iz razloga što znatno olakšava pisanje nekih inače kompliciranih struktura kao što su reference, fusnote, indeksi, popisi slika i sl. U  $\LaTeX$ -u se za upis literature koristi ili okruženje `\thebibliography` ili zasebni *bibtex* (.bib) dokument. Bibtex je dokument u kojem se upisuju svi citirani radovi, a naknadno se on poziva preko naredbe `\bibliography{nazivdokumenta}`. U njega se upisuju detalji o radu na ovaj način :



```

@article{einstein,
  author =      "Albert Einstein",
  title =      "{Zur Elektrodynamik bewegter K{\\"o}rper}.
               [{On} the electrodynamics of moving bodies]",
  journal =    "Annalen der Physik",
  volume =    "322",
  number =    "10",
  pages =     "891--921",
  year =      "1905",
  DOI =       "http://dx.doi.org/10.1002/andp.19053221004"
}

```

U tekstu se naznači citat na ovaj način citirani tekst `\cite{einstein}`  
 Kod thebibliography okruženja nešto se drugačije navodi.

```

%tekst koji želimo citirati
Posebno je važno je da se \TeX ime piše pravilno kako bi se
razlikovao od TEX (izgovara se \textit{teks}) akronima za Text
EXecutive processor. \cite{dknuth} %naredba za citiranje

%naredba za stvaranje bibliografije
\begin{thebibliography}{}
\bbitem{dknuth}
Donald E. Knuth. (1984) \textit{The TeXbook, Volume A of
Computers and Typesetting},
Addison-Wesley Publishing Company, ISBN 0-201-13448-9.
\end{thebibliography}

```

U tom okruženju nije potreban vanjski dokument, a naziv reference oblikuje se ručno.  
 Kako to izgleda kada se generira dokument prikazano je na slikama 11 i 12.  
 Naredbi `\cite{naziv}` u tekstu koji citiramo moramo pridodati oznaku *naziv* preko  
 koje ćemo kasnije u bibliografiji spojiti taj citat s literaturom.

Posebno je važno je da se  $\TeX$  ime piše pravilno kako bi se razlikovao od TEX (izgovara se teks) akronima za Text EXecutive processor. [11]

Slika 11. Izgled citata u tekstu

[11] Donald E. Knuth. (1984) *The TEXbook, Volume A of Computers and Typesetting*, Addison-Wesley Publishing Company, ISBN 0-201-13448-9.

Slika 12. Izgled reference u literaturi

U `\begin{thebibliography}` okruženju pozivamo citate preko naredbe `\bibitem{naziv}`.

Da bi se popis literature pojavio u sadržaju moramo ga nekako spojiti. Za to se u *unizg* klasu ubacio još jedan paket, `\RequirePackage{tocbibind}` koji automatski spaja bibliografiju, indeks, listu slika i slično s popisom sadržaja. U ovom slučaju moralo se pridodati i funkcija koja u isključuje sam sadržaj iz sadržaja, `\RequirePackage[nottoc]{tocbibind}`.

### 3.5. Pripremanje *unizg\_klasa.cls* za postavljanje na CTAN i GitHub

Odlika online  $\LaTeX$  distribucija jest da se lako surađuje s drugim autorima te je razmjena dokumenata puno jednostavnija. No kako je na početku ovog diplomskog rada određeno, ova klasa bit će postavljena i u konvencionalnom načinu na internet kako bi bila dostupna i onima koji ne koriste Overleaf. U tom slučaju "pakiranje" klase je nešto kompliciranije. Uobičajeno je sve stilove (.sty) ili pakete te klase(.cls) distribuirati kao .dtx datoteku. Dtx datoteka skuplja sve relevante dokumente u jedan, sadrži komentirani izvorni kod i dokumentaciju u jednom. Da bi se takva datoteka uopće mogla otvoriti mora se stvoriti i instaler, odnosno .ins datoteka. To je vrlo kratki kod koji pomaže izvući sve sadržano u .dtx datoteci. Takve dvije datoteke mogu otvoriti svi  $\LaTeX$  korisnici i ne razlikuju se znatno od obične .tex datoteke ali u sebi sadrže uputu za  $\LaTeX$ .

Nakon što je stvorena .dtx i .ins datoteka može započeti *upload* na CTAN i GitHub. Oda-

brana su ta dva portala iz razloga što su najpopularniji za dijeljenje  $\LaTeX$  datoteka. Proces postavljanja nije naročito kompliciran. Prvi korak bio je stvoriti korisnički račun, a zatim upload preko gumba *New project* ili *New repository*. Klasa je javno dostupna na <https://www.ctan.org> i <https://github.com/MateaMilin/unizgclass> pod licencom LaTeX Project Public License. Moguće ju je pronaći pod nazivom *unizgclass* te ju je dopušteno uređivati i ponovo distribuirati pod uvjetom da se promijeni ime kako ne bi došlo do zabune. Na održavanju klase, osim autorice, radit će i izv. prof. dr. sc. Ante Poljićak.

### 3.6. Korištenje *unizg\_klasa.cls*?

Kada se skine *unizg\_klasa* sa CTAN-a ili GitHub-a biti će u .zip formatu. Potrebno ga je izvesti kako bi se moglo pristupiti mapi s datotekama. Mapa će sadržavati *readme.mn*, *unizg\_klasa.dtx* i *unizg\_klasa.ins*. U *readme.mn* datoteci sadržane su upute za instalaciju. Prvo se pokreće *unizg\_klasa.ins* datoteka koja omogućava pokretanje .dtx datoteke na način da se u nekom od editora, TeXShop, TeXMaker i sl., klikne pokreni (*typeset*). Nakon toga može se pokrenuti .dtx datoteka koja će stvoriti *example.tex* i *unizg\_klasa.cls*. Datoteku *unizg\_klasa.cls* potrebno je kopirati u direktorij gdje se nalazi .tex datoteka koju se trenutno piše.

Kada se pozove *unizg\_klasa.cls* datoteka pomoći `\LoadClass` važno je u glavnoj .tex datoteci ispuniti tokene za pravilno generiranje dokumenta. Točan način prikazan je u *example.tex* datoteci. Postoji 8 tokena koji se pozivaju nakon `\begin{document}`:

```
\imestudenta{ime i prezime}
\sveuciliste{naziv sveučilišta}
\fakultet{naziv fakulteta}
\naslovrada{naslov rada}
\mentor{ime, prezime i titula mentora}
\rad{vrsta rada, diplomski}
\grad{npr. zagreb}
% nakon toga se pozivaju naredbe
\prvanaslovna
\druganaslovna
```

## 4. REZULTATI I RASPRAVA

Kao rezultat ovog rada je izrađena  $\LaTeX$  klasa (.cls datoteka) te kratke upute za korištenje programa  $\LaTeX$ . Upravo ovaj diplomski rad je oblikovan korištenjem *unizg* klase. Za izradu klase korišten je  $\LaTeX$  opisni jezik u online programu za pisanje  $\LaTeX$  dokumenata - Overleaf. Velik dio vremena utrošen je na proučavanje principa rada  $\TeX$ -a i  $\LaTeX$ -a. S obzirom na to da znanje i rad u navedenim programima na početku ovog rada nije postojalo te da se iz sličnih programskih jezika poznavala samo osnova, najveći izazov bio je naučiti pozadinu rada programa. Drugi veliki zadatak jest bio savladati kako točno adresirati argumente u kodu te da se oni pravilno izvrše. Kako se pretpostavlja da novu  $\LaTeX$  klasu neće pisati potpuni početnik bilo je teško snalaziti se u stručnim pojmovima vezanih uz program. To se naročito odnosi na razumijevanje tzv.  $\TeX$  primitiva odnosno osnovnih naredbi kao što je `\def` ili `\newtoks`. Jedan od problema jest i golem broj rješenja za jedan problem u okviru raznih novih paketa koji se mogu naći na internetu. Također ne postoji jedan priručnik s najčešćim problemima jer se cijeli sustav brzo mijenja i nova rješenja izlaze skoro svakodnevno iz kog razloga knjige i priručnici brzo zastarijevaju. Na početku rada krenulo se s pretpostavkom da je pisanje u  $\LaTeX$  znatno jednostavnije od pisanja u nekom od WYSIWYG programa. Ispostavilo se da je to djelomično točno. Ono što se iz osobnog iskustva naučilo jest da  $\LaTeX$  eliminira sve frustrirajuće faktore koje WYSIWYG posjeduju, kao ubacivanje slika, numeracije, razdvajanje na poglavlja, definiranje zaglavlja itd. te da je neusporedivo jednostavniji za oblikovanje dugačkih dokumenata kao što su knjige koje imaju i po 500 stranica. No i kod  $\LaTeX$ -a postoje mane, a to su pristup oblikovanju teksta koji nije intuitivan, potpuno nov način pisanja sintaksi, korištenje specijalnih znakova koji se često javljaju i u tekstu pa ih se mora posebno adresirati, često javljanje greški i pronalaženje novih paketa za rješavanje minornih grešaka. Ipak rezultat je zadovoljavajuć te otvara nove mogućnosti za izradu uniformirane klase za sva hrvatska sveučilišta. Naročito bi pomogao studentima prirodnih znanosti (matematika, fizika, kemija) i tehničkih usmjerenja koji često u svojim radovima moraju pisati kompleksne formule i kodove. Vidi se i prostor za grafičke dizajnere i urednike koji bi preko  $\LaTeX$  mogli stvarati predloške za oblikovanje izvješća i prezentacija u velikim kompanijama.

## 5. ZAKLJUČAK

Istražujući teorijsku pozadinu rada te uspješna izrada klase *unizg\_klasa.cls* dovela je do zaključka da je  $\LaTeX$  sistem za oblikovanje dokumenata odličan izbor pri oblikovanju dugih dokumenata s kompliciranom strukturom i zahtjevnim matematičkim ili drugim specijalnim znakovima. Potvrđene su sve 3 hipoteze navedene u radu.  $\LaTeX$  omogućava korisniku da se usredotoči na sadržaj rada jer ga ne opterećuje s oblikovanjem teksta i umetanjem slika što rezultira većim utroškom vremena u kvalitetu sadržaja. U  $\LaTeX$ -u se uz pomoć raznih paketa, stilova i klasa može postići skoro svaki dizajn koji se može postići i u Microsoft Word-u, a dokaz za to su razne klase i dizajn za oblikovanje prezentacija, životopisa, knjiga, radova i sl. na stranicama GitHub, CTAN i Overleaf, što potvrđuje drugu hipotezu ovog rada. Također je potvrđena i treća hipoteza ovog rada -  $\LaTeX$  je jednostavan za korištenje, odnosno može ga koristiti prosječna osoba bez predznanja u programu. To se vidi iz samog oblikovanja ovog rada gdje je autorica bez predznanja uspjela uspješno oblikovati običan .tex dokument, ali i izraditi potpuno novu klasu.

Unatoč svi potvrđenim tezama vidi se mjesto za napredak  $\LaTeX$ -a što se i očekuje u odavno najavljenom  $\LaTeX$ 3 izdanju. Prvenstveno se to očituje kod pisanja običnog dokumenta gdje se za svaku promjenu i sitnicu mora učitati novi paket, što korisnici i developeri smatraju prvim problemom.

Klasa koja je nastala kao rezultat ovog diplomskog rada pokazala se nešto zahtjevnijom od običnog pisanja u  $\LaTeX$  te je odgovor na probleme u tom segmentu teže naći. No, ipak ako je nužno moguće je napisati klasu i sa slabijim predznanjem, svakako se ona naknadno kako se otkrivaju problemi može ažurirati. Pokazalo se da je u tom slučaju online distribucija Overleaf znatno jednostavnija od korištenje lokalnih distribucija. Sama činjenica da nije potrebna instalacija programa na računalo daje prednost Overleaf-u, a kada su u pitanju učitavanje paketa i klasa, dijeljenje koda, grafičko sučelje i podrška kod sistemskih grešaka on potpuno zasjenjuje lokalne distribucije.

Realizirana klasa *unizg\_klasa.cls* se vrlo lako može ažurirati i unaprijediti po potrebama te će se budno pratiti svaki zahtjev budućih korisnika, a nije isključeno ni da će se, ukoliko se pokaže interes, potpuno iznova izraditi uniformirana klasa za sve sastavnice Sveučilišta u Zagrebu.

## 6. Literatura

- [1] *Poor quality typesetting on the second edition of my book*, dostupno na:  
<https://www.webofstories.com/play/donald.knuth/50>, 15.3.2019.
- [2] Knuth D. E. (1990). *The Future of T<sub>E</sub>X and METAFONT*, dostupno na:  
<http://www.ntg.nl/maps/05/34.pdf> , 15.3.2019
- [3] *Donald (Don) Ervin Knuth*, dostupno na:  
[https://amturing.acm.org/award\\_winners/knuth\\_1013846.cfm](https://amturing.acm.org/award_winners/knuth_1013846.cfm), 14.5.2019.
- [4] *Knuth reward check*, dostupno na:  
[https://en.wikipedia.org/wiki/Knuth\\_reward\\_check](https://en.wikipedia.org/wiki/Knuth_reward_check), 15.6.2019.
- [5] Pégourié-Gonnard M. (2013). *A guide to Lua $\mathcal{E}$ T<sub>E</sub>X*, dostupno na:  
<http://ctan.ijs.si/tex-archive/info/luatex/lualatex-doc/lualatex-doc.pdf> , 15.3.2019.
- [6] Mittelbach F. (2006). *Interview of Frank Mittelbach: A combined interview of the  $\mathcal{E}$ T<sub>E</sub>X Project director, Free Software Magazine*, 11 (0), dostupno na:  
<https://www.latex-project.org/publications/2006-FMi-FSM-interview.pdf> ,  
15.3.2019.
- [7] Ungar Š. (2002.) *Ne baš tako kratak Uvod u T<sub>E</sub>X s naglaskom na  $\mathcal{E}$ T<sub>E</sub>X 2<sub>ε</sub>: ili  $\mathcal{E}$ T<sub>E</sub>X 2<sub>ε</sub> u xii + 107 minuta i 56 sekundi*, Sveučilište J.J. Strossmayera u Osijeku, Odjel za matematiku, Osijek
- [8] *Getting  $\mathcal{E}$ T<sub>E</sub>X*, dostupno na: <https://www.latex-project.org/get/> 16.3.2019.
- [9] L<sup>a</sup>mport L. (2018) *My writings*, dostupno na:  
<http://lamport.azurewebsites.net/pubs/pubs.pdf>, 16.3.2019.
- [10] Rowley C. (2001). *The  $\mathcal{E}$ T<sub>E</sub>X Legacy, 2.09 And All That*, dostupno na:  
<https://www.latex-project.org/publications/2001-CAR-LaTeX-legacy.pdf>,  
16.3.2019.

- [11]  $\text{\TeX}$ ,  $\text{\LaTeX}$  and  $\text{\AMS-LaTeX}$ , dostupno na :  
<https://web.archive.org/web/19981203013048/http://web.syr.edu:80/~trscavo/latex/latex-history.html>, 16.3.2019.
- [12] Lamport L. (1994). *\LaTeX: A Document Preparation System*, Addison-Wesley Publishing Company, Reading, Massachusetts, second edition, ISBN 0-201-52983-1.
- [13] Donald E. Knuth. (1984) *The \TeXbook, Volume A of Computers and Typesetting*, Addison-Wesley Publishing Company, ISBN 0-201-13448-9.
- [14] *What is CTAN?*, dostupno na : <https://ctan.org/ctan/>, 17.3.2019.
- [15]  $\text{\LaTeX}$ 3 projektni tim (2006). *\LaTeX 2 $\epsilon$  for class and package writers*, dostupno na :  
<https://www.latex-project.org/help/documentation/clsguide.pdf>, 1.5.2019.
- [16] Oetiker T., Partl H., Hyna I., Schlegl E. (2018.) *The Not So Short Introduction to \LaTeX 2 $\epsilon$  Or \LaTeX 2 $\epsilon$  in 139 minutes*, dostupno na:  
<https://tobi.oetiker.ch/lshort/lshort.pdf>, 1.5.2019.
- [17] *A six-part series: How do TeX macros actually work?*, dostupno na:  
[https://www.overleaf.com/learn/latex/A\\_six-part\\_series:\\_How\\_do\\_TeX\\_macros\\_actually\\_work%3F](https://www.overleaf.com/learn/latex/A_six-part_series:_How_do_TeX_macros_actually_work%3F) 24.6.2019.



# A. DODATAK

## A.1. Kratke upute za korištenje $\LaTeX$ -a

U ovom diplomskom radu cilj je bio pomoći studentima, budućim korisnicima  $\LaTeX$ -a, profesionalno oblikovati njihove radove. S obzirom na to da pisanje u  $\LaTeX$ -u zahtijeva poznavanje *markup* jezika, prije početka važno je savladati osnovne  $\LaTeX$  naredbe koje pomažu u oblikovanju diplomskog rada, seminara i članka te bez kojih je kvalitetan rad teško zamisliti. Kako se ovaj rad ipak veže uz izradu klase neće se previše duljiti, opširnije upute mogu se pročitati u knjigama *The  $\LaTeX$  Companion* F. Mittelbacha,  *$\LaTeX$ : A Document Preparation System* L. Lamporta, *Ne baš tako kratak Uvod u  $\TeX$  s naglaskom na  $\LaTeX 2_{\epsilon}$*  Š. Ungara ili na mnogobrojnim online izvorima kao što je Overleaf.com, tex.stackexchange.com, CTAN.org i mnogi drugi.

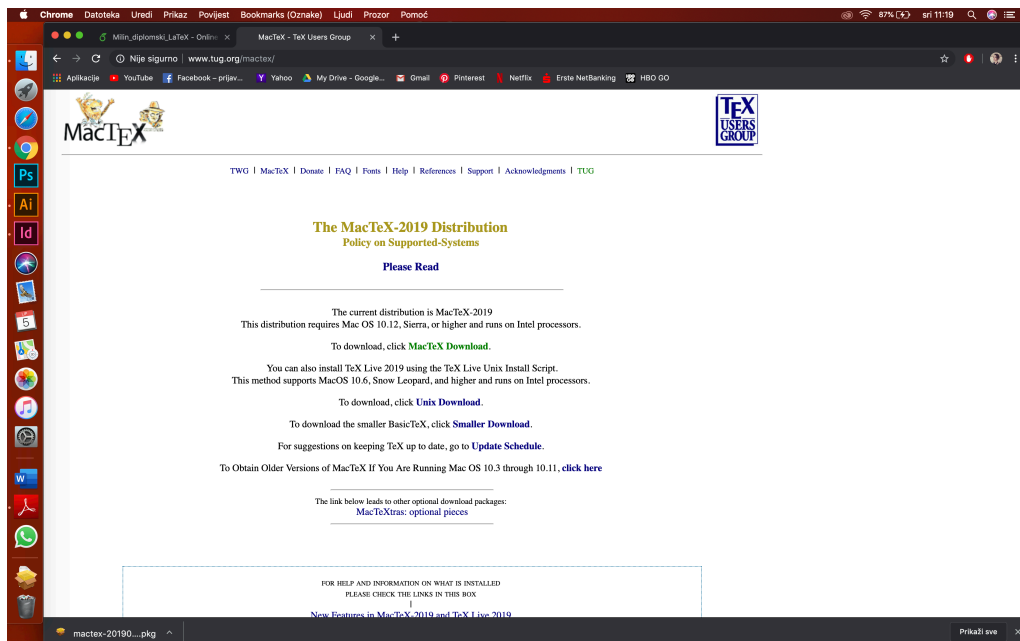
## A.2. *Download* i instalacija

Da bi se  $\LaTeX$  mogao koristiti na bilo kojem računalu potrebno je imati nekoliko programa:

1.  $\TeX$  ili  $\LaTeX$  program za pretvaranje izvornih  $\LaTeX$  dokumenata u PDF ili DVI dokument.
2. Program za uređivanje teksta u kojem će se pisati  $\LaTeX$  kod (npr. TeXShop)
3. PDF/DVI program za pregledavanje i tiskanje dokumenata (npr. Adobe Reader)
4. Program za upravljanje i ubacivanje POSTSCRIPT dokumenata i slika u dokument [16]

$\LaTeX$  je besplatan software pod javnom licencom  $\LaTeX$  projekta (LPPL).  $\LaTeX$  se primarno distribuira preko CTAN servera ili mnogih  $\TeX$  distribucija koje omogućava  $\TeX$  User Group (TUG). Dobar početak za pronalaženje odgovarajuće  $\LaTeX$  distribucije je službena web stranica  $\LaTeX$  projekta, <https://www.latex-project.org/get/>.

Za potrebe pisanja ovog rada odabrana je online verzija  $\LaTeX$ a - Overleaf, zbog jednostavnosti i djelomičnog *built-in* prikaza PDF-a te zbog jednostavne komunikacije i lakog



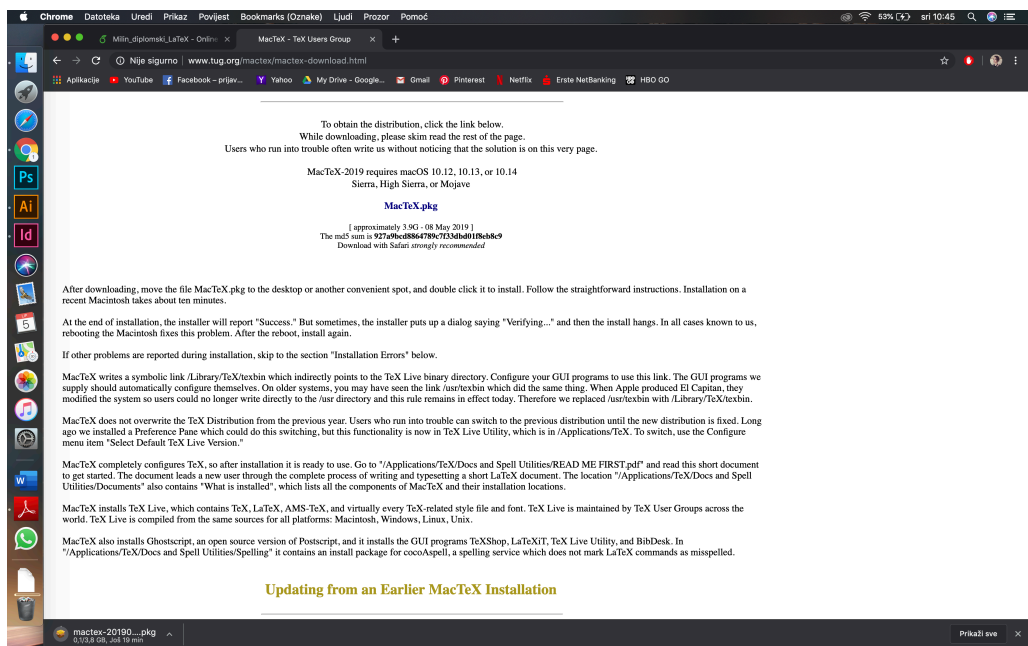
Slika 13. Web stranica za *download* MacTeX-a

dijeljenja s ostalim korisnicima. No, zbog toga što većina  $\text{\LaTeX}$  korisnika koristi neki od računalnih programa, kao što su MiKTeX, proTeX i drugi, i eksperimentalnog karaktera ovog diplomskog rada važno je bilo isprobati i usporediti oba načina generiranja  $\text{\LaTeX}$  dokumenta. Za računalni program odabrana je  $\text{\LaTeX}$  distribucija sa službene web stranice  $\text{\LaTeX}$  projekta - MacTeX. Važno je odabrati program koji odgovara operativnom sustavu, u ovom slučaju to je MacOS pa je zato odabran MacTeX. Web link na službenoj stranici vodi TUG stranicu prikazanu na slikama 13 i 14, s koje se jednim klikom na "To download, click MacTeX Download" pokreće skidanje MacTeX paketa. Prikazana procedura se odnosi na MacOS operativni sustav, za Windows operativni sustav moglo bi biti neznatno drugačija instalacija.

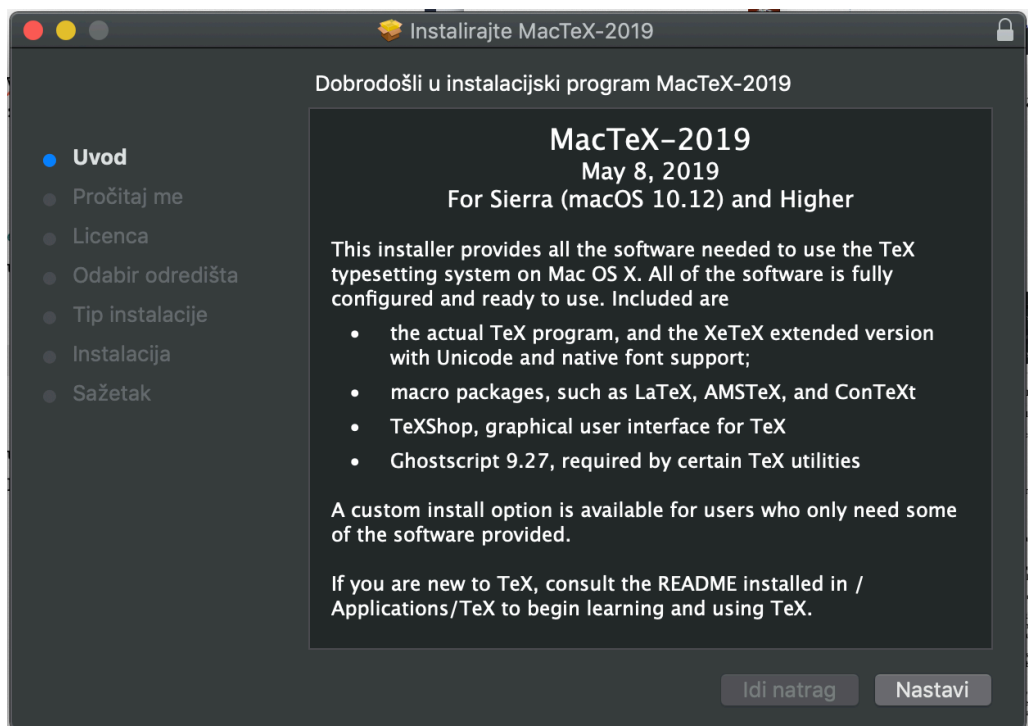
Klikom na gumb za *download* otvara se stranica s uputama za instalaciju prikazana na slici 14.

Kada skidanje paketa završi potrebno je otvoriti .pkg datoteku i prikazuje se izbornik kao na slici 15. Potrebno je proći sve navedene korake i prihvatiti Uvijete korištenja. Program zauzima približno 7 gigabajta prostora na računalu.

Nakon instalacije na računalu će se moći naći programi LaTeXIt, TeXShop, BibDesk i TeX Live Utility. BibDesk je program koji olakšava citiranje i upravljanje bibliografijom, TeX Live Utility je program za provjeru ažuriranja TeXShop-a, TeXLive-a i instaliranih



Slika 14. Upute za instalaciju



Slika 15. Instalacija paketa

paketa, a LaTeXiT je program u kojem je pisanje matematičkih formula znatno olakšano i služi prvotno tome. Ono što je zapravo  $\LaTeX$  distribucija jest TeXLive.

TeXShop je zapravo program za upisivanje  $\LaTeX$  koda dok ga zapravo u pozadini pokreće TeXLive. Kada otvorimo TeXShop automatski se otvara i početni dokument *Untitled.tex*. Sljedeći koraci opisani su u nastavku.

### A.3. $\LaTeX$ distribucije, *engines* i editori

Početniku  $\LaTeX$  izrazi mogu biti izrazito zbunjujući, naročito kada se to odnosi na *software* koji su potrebni za normalan rad. Razlikuju se *systems*, *distributions*, *engines* i *editors*.

Sistemi za oblikovanje su veliki paketi makro naredbi koji izvorne  $\TeX$  naredbe preoblikuju da budu logički više strukturirani i jednostavniji za korištenje. Osim  $\TeX$ a tu su još AMSTeX, ConTeXt i  $\LaTeX$  opisani u tablici 2. Danas najkorišteniji je ipak  $\LaTeX$ . Kodovi napisani za različite sisteme najčešće međusobno nisu kompatibilni.

Distribucije (*distributions*) su skup paketa i programa koji uključuju kompajlere, fontove i makropakete. One omogućavaju izradu dokumenata bez ručnog upravljanja datotekama. One distribuiraju sve nužno za funkcioniranje računalnog mehanizama  $\TeX$ .

Računalni mehanizmi (*engines*) ili kako ih još nazivaju *compilers* su izvršne datoteke koje pretvaraju izvorni kod u izlazni format za ispis. Oni obrađuju samo sintakse te učitavaju fontove i makronaredbe da bi uopće razumjeli kod i točno generirali izlaznu datoteku. Razlika između računalnih mehanizama napisana je u tablici 2.

Editori su programi za obradu teksta i izvornog koda u kojima nije moguće mijenjati izgled generiranog dokumenta te se koriste uglavnom za izmjenu programskog koda u npr. HTML-u- Bitno se razlikuju od tekst procesora koji oblikuju, formatiraju i generiraju dokument, npr. Microsoft Word. Tekst procesori se rijetko koriste za izmjenu koda. Popis  $\LaTeX$  editora nalazi se u tablici 2.

Tablica 2. Popis distribucija, sistema, kompajlera i editora

<b>Sistem</b>	<b>Opis</b>
AMSTeX	Program za obradu dokumenata korišten od strane Američkog matematičkog instituta (AMS) od 1982. do 1985. Danas u obliku AMS $\LaTeX$ koji sadrži <i>amsmath</i> paket koji se koristi u skoro svakom $\LaTeX$ dokumentu.
ConTeXt	Program za obradu dokumenata kojeg su osmislili Hans Hagen i Ton Otten 1991. godine. ConTeXt pretpostavlja da je autor teksta ujedni i grafički urednik te zbog toga ima nešto jednostavnije i lakše razumljive sintakse.
$\LaTeX$	Program za obradu dokumenata. Pretpostavlja da autor i grafički urednik nisu ista. Iz tog razloga je kompleksniji i postoje tzv. klase dokumenata koje daju izvrsno . generirane dokumente.
<b>Distribucija</b>	<b>Opis</b>
TeX Live	Distribucija neovisna o platformi
MacTeX	Distribucija namijenjena MacOS, bazirana na TeX Live distribuciji
MiKTeX	Distribucija namijenjena Windows operativnom sustavu
<b>Računalni mehanizam</b>	<b>Opis</b>
xetex, xelatex	$\TeX$ mehanizam koji podržava Unicode fontove, True Type Font i Open Type Font
LuaTeX	$\TeX$ mehanizam koji ima uključenu podršku za Lua jezik, ns ciljem da doda fleksibilnost $\TeX$ u. Podržava Unicode fontove.
pdfTeX, pdfLaTeX	$\LaTeX$ koji generira PDF dokumente
$\TeX$ , $\LaTeX$	Izvorni mehanizmi koji generiraju DVI dokumente.
<b>Programi za uređivanje teksta (editors)</b>	<b>Opis</b>

BoKoMa	TeX editor koji funkcionira na 'What You See Is What You Get' principu. Rezultati se prikazuju u realnom vremenu. Podržavaju ga MacOS, Windows i Linux operativni sustavi.
LyX	Editor koji funkcionira na 'What You See Is What You Mean' principu. Autor je potpuno lišen L <sup>A</sup> T <sub>E</sub> X sintaksi i ne kontrolira ih. Za ono što nije podržano u grafičkom korisničkom sučelju ipak je potrebno poznavanje L <sup>A</sup> T <sub>E</sub> X-a. U realnom vremenu prikazuje djelomični izgled finalnog dokumenta. Podržan je na svim operativnim sustavima.
TeXMaker	Jedan od najčešće korištenih editora s dobrim grafičkim sučeljem koji je dostupan na svim operativnim sustavima.
TeXstudio	<i>Open-source</i> editor dostupan na svim platformama, izdvojen iz TeXMakera.
TeXWorks	Editor razvijen da bude što jednostavniji kako bi se početnici lakše snašli. Dostupan na svim operativnim sustavima.
Notepad ++	Tekst editor koji dolazi sa svim Windows računalima. Nema integrirani PDF preglednik.
Overleaf	Online editor, siguran i jednostavan za korištenje i dijeljenje. Sadrži parcijalni WYSIWYG preglednik. Sada je spojen s nekadašnjim ShareLaTeX-om
CoCalc	Online stranica za dijeljenje i surađivanje u računanju koja nudi i editor za L <sup>A</sup> T <sub>E</sub> X dokumente.

## A.4. Početak rada

### A.4.1. Upisivanje naredbi i komentari

Svaka naredba u L<sup>A</sup>T<sub>E</sub>X-u započinje sa *backlash* znakom `\` zatim, bez razmaka, naredbom. Naredba završava razmakom ili znakom koji nije slovo. Kod nekih specifičnih naredbi L<sup>A</sup>T<sub>E</sub>X zanemaruje razmake iza naredbi. Ako je potrebno dobiti razmak iza naredbe mora se upisati `{ }` pa zatim razmak. Najčešće je to slučaj kod upisivanja službenog TeX i L<sup>A</sup>T<sub>E</sub>X

loga gdje će se, ako ne upišemo {}, logo spojiti sa sljedećom riječi u rečenici.

Nastavno na to, neovisno koliki broj razmaka upišemo  $\LaTeX$  će i dalje prikazati samo jedan. Za započeti novu rečenicu, odnosno paragraf, u novom redu potrebno je ostaviti praznu liniju koda između njih. Primjer takvog koda:

```
Ovo je primjer da će  $\LaTeX\{}$  bez obzira na količinu  
↔ razmaka uvijek prikazati samo jedan razmak.
```

```
Ova rečenica je primjer  
da "enter" ne  
djeluje kao novi red u  $\LaTeX\{}$ u.
```

```
Osim ako se ne stavi barem jedna prazna  
linija koda.
```

---

```
Ovo je primjer da će  $\LaTeX$  bez obzira na količinu razmaka uvijek prikazati samo  
jedan razmak. Ova rečenica je primjer da "enter" ne djeluje kao novi red u  $\LaTeX$ u.  
Osim ako se ne stavi barem jedna prazna linija koda.
```

Sve napisano u istom retku kao i znak % smatra se komentárom što je dobro za prikazivanje teksta koji ne želimo da se vidi u završnom dokumentu, te za vizualno odvajanje teksta od koda i sl. Ako nam je znak % potreban u tekstu potrebno je upisati \% kako bi se on prikazao. Isto vrijedi i za znakove #, \_, \$, &, {, }. Razlog tome je što su to specijalni  $\TeX$  znakovi koje on koristi kod definiranja naredbi. Također znak \ se upisuje na način `\textbackslash`.

#### A.4.2. Pozivanje klase

Da bi  $\LaTeX$  mogao izgenerirati nekakav dokument potrebno je pozvati neku od klasa koja oblikuje naš dokument. Najčešće su to *article*, *letter*, *report* ili *book*. Nakon pozivanja klase započinje se dokument naredbom

```
\begin{document}
```

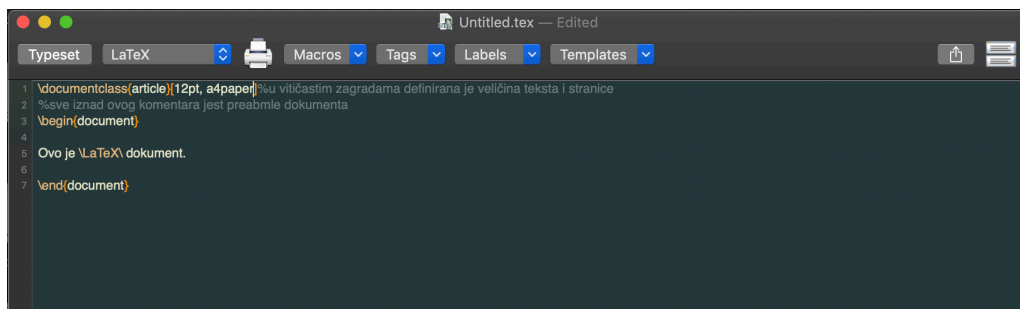
Kako bi  $\LaTeX$  mogao izgenerirati dokument važno je zatvoriti dokument naredbom

```
\end{document}
```

Sve ono što želimo da se ispiše, vidi i prikaže mora se upisati unutar tih dviju naredbi. Tako bi osnovni kod, koji je moguće izgenerirati, s nekim tekstom izgledao ovako:

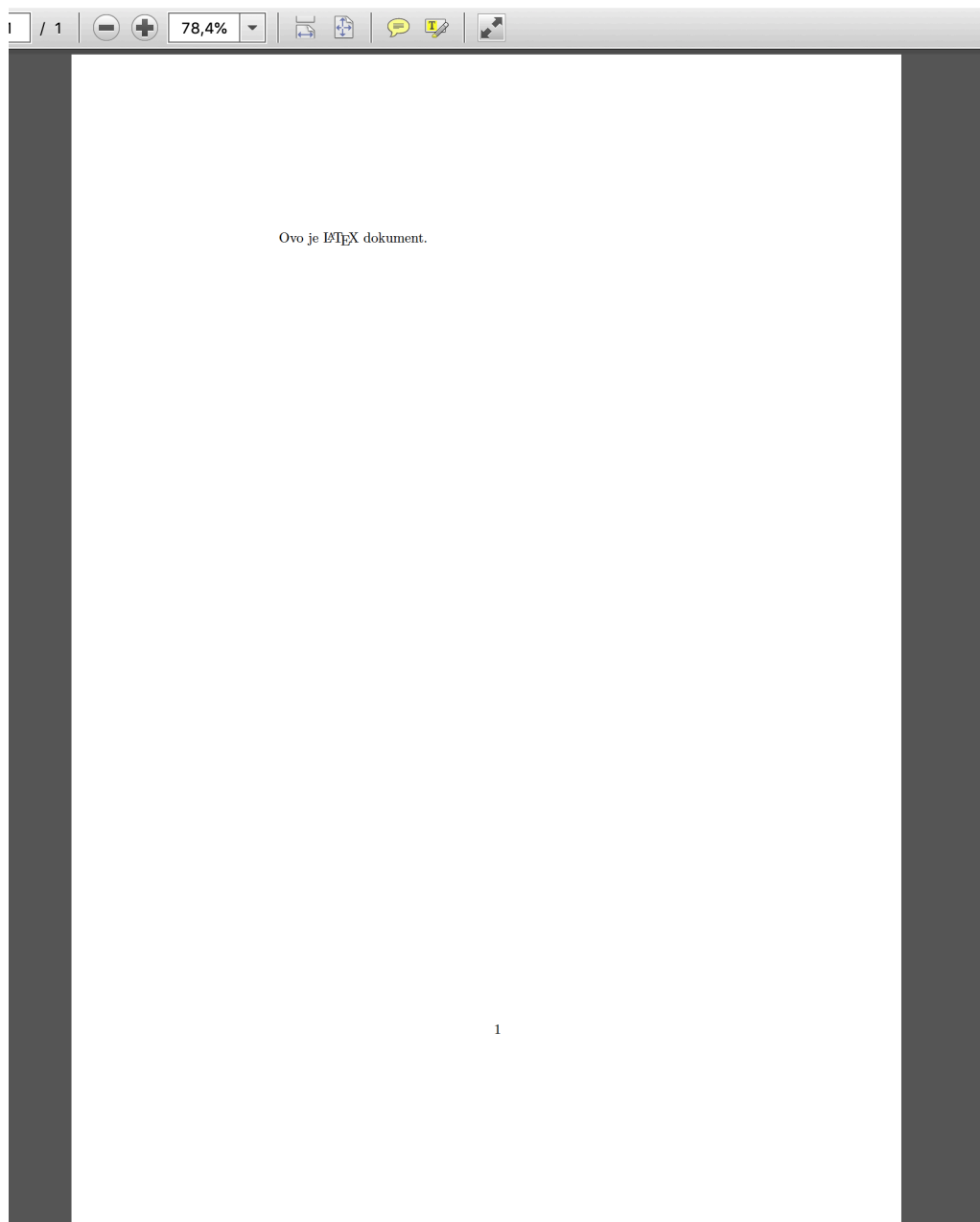
```
\documentclass{article}
\begin{document}
Ovo je primjer rečenice napisane u LaTeXu
\end{document}
```

Ono što određuje izgled dokumenta i sve upute za  $\text{\LaTeX}$  o formatiranju teksta upisuje se u tzv. *preamble* dokumenta. *Preamble* je dio koda koji se nalazi prije `\begin{document}`. U  $\text{\LaTeX}$  u uglate zagrade „`[]`” upisuju se opcionalne naredbe, a u vitičaste „`{}`” obavezne bez kojih naredba neće funkcionirati. Slika 16 primjer je kako izgleda jednostavan  $\text{\LaTeX}$  kod, a slika 17 kakav dokument on generira.



Slika 16. Primjer jednostavnog koda u programu TeXShop





Slika 17. Izgenerirani kod

## A.5. Korištenje paketa

Osnovni  $\LaTeX$  ne može riješiti sve probleme pa se često mora koristiti niz dodatnih paketa da bi se prikazalo sve što želimo. Ukoliko želimo ubaciti sliku, tablicu ili koristiti tekst u boji treba koristiti takozvane makro pakete ili *packages*. Njih aktiviramo naredbom `\usepackage [opcije] {makro paket}` koja se upisuje u *preamble* dokumenta. Pri tome je makro paket naziv željenog paketa, a opcije su niz ključnih riječi s kojima dodatnom možemo upravljati paketom.[7]

Neki osnovni paketi dolaze s  $\LaTeX$  distribucijom no glavninu paketa ipak se može pronaći na gore spomenutom CTAN-u ili GitHub-u. Upute za instalaciju i korištenje te detaljan opis dolaze s paketom. Paketi najčešće dolaze u `.dtx` ili `.ins` formatu iz kojih se mora "izvući" željeni stil. Obično se to radi tako da se dokument s `.ins` ekstenzijom "provuče" kroz  $\LaTeX$  odnosno generira u nekom od programa kao što je TeXShop. Nakon toga se dobivene datoteke `.sty` i `.ltd.tex` kopiraju u istu mapu gdje se nalazi i `.tex` datoteka.

## A.6. Uređivanje dokumenta i osnovne naredbe

### A.6.1. Postavljanje margina i dimenzije

Ponešto o marginama je već napisano u poglavlju 3.2.1 Dimenzije i margine. Dimenzija dokumenta može se definirati pri pozivu klase, npr.

```
\documentclass[11pt,a4paper,twocolumn]{article}
```

Naravno svaka klasa ima neke svoja ograničenja, a u slučaju `article` klase to su da veličina teksta, bez uvođenja nekih od paketa, može biti 10, 11 ili 12 pt. U ovom primjeru može se vidjeti da se kod `article` klase može odmah definirati da tekst bude prelomljen u 2 stupca.

Da bi uveli dimenzije stranice koje nisu konvencionalne najbolje je koristiti neki od paketa kao što su `geometry`, također se s njim može izmjeniti veličina margina kao što je u primjeru ove unizg klase, prikazano u poglavlju 3.2.1.

## A.6.2. Jezik i gramatika

$\LaTeX$  2 $\epsilon$  je trenutno standardna verzija koju koristi većina korisnika kojoj je standardni jezik engleski.  $\LaTeX$  koristi ASCII znakove zbog kojih ne može prikazivati UTF-8 znakove kao što su hrvatski dijakritički znakovi. Ipak njih se može prikazati na nekoliko načina, npr. `\v C`, no takav pristup sada je već zastario i postoje jednostavniji načini. Paket `\usepackage{inputenc}{utf-8}` rješava problem hrvatskih znakova. U slučaju da se koristi XeLaTeX *enigine* čak ni to nije potrebno.

Isto tako, engleski jezik ima drugačije rastavljanje riječi od hrvatskog pa se za pravilniju rastavljanje koristi paket `\usepackage[croatian]{babel}`.

## A.6.3. Naslovi i podnaslovi

Pisanje podnaslova i naslova koji se automatski povezuju sa sadržajem u  $\LaTeX$ -u je vrlo jednostavno. Naslov je `\section{}`, podnaslov `\subsection{}`, podpodnaslov je `\subsubsection{}`, te još postoji poglavlje u *book* klasama `\chapter{}` i odlomak `\paragraph{}`.

## A.6.4. Naglašavanje teksta

U  $\LaTeX$ -u postoje 4 osnovne vrste naglašavanja riječi. To su:

- `\textbf{tekst}` za definiranje **bold** riječi,
- `\textit{tekst}` za definiranje *italic* riječi,
- `\underline{tekst}` za definiranje podcrtanih riječi,
- `\emph{tekst}` za definiranje *naglašenih* riječi. Kod `\emph` naredbe tekst je također *italic*, no ako unutar te naredbe još jednom definiramo *emph* ta riječ biti će u normalnom obliku

## A.7. Ubacivanje slika i Graphicx paket

$\LaTeX$  sam po sebi ne može upravljati slikama pa se za njih mora uvesti paket `\usepackage{graphicx}`. Kako bi uopće mogao pronaći mora se definirati točna loka-

cija slika s naredbom `\graphicspath{}`

koja se upisuje u preambulu dokumenta. Samo ubacivanje slike obavlja se preko naredbe `\includegraphics [] {}` Važno je točno adresirati mapu u kojoj se nalaze slike na jedan od ovih načina:

```
\graphicspath{{primjer/}} %slike se nalaze u mapi "primjer"
%u istoj mapi gdje je i .tex datoteka koja sadrži includegraphics
\graphicspath{{./primjer/}}% Ako slučajno kad u istoj
% mapi ima više .tex datoteka
%na taj način je putanja vezana uz glavni .tex dokument
\graphicspath{{C:/korisnik/primjer}}%apsolutna putanja
%koja točno adresira folder
\graphicspath{{./primjer1/}{./primjer2/}}
%ako ima više mapa sa slikama
```

Slike se mogu ubacivati i preko okruženja *figure* koje spada u tzv. *float* okruženje odnosno elementi koji mogu putovati. Takvo okruženje omogućava da  $\LaTeX$  sam izabere najbolje moguće mjesto za pozicioniranje slike. Slijedi primjer:

```
\begin{figure}[!ht]
  \centering
  \captionsetup{justification=centering}
  \includegraphics [height=6cm]{Slike/slika16.jpeg}
  \caption{\textit{Dhalia}, Janine Joles,
  \url{https://unsplash.com/photos/f0heeu-Ec0}}
  \label{primjer}
\end{figure}
```

Ovaj kod dao je rezultat da se slika 18 prikazuje ovako:

Naredbom `\begin{figure}` započelo je stvaranje slike te se zadala uputa `!ht` što znači `h=here` neka slika ostane ovdje, `t=top` ili neka bude na vrhu stranice, te `!=` izvedi `ht` i ako rezultati ne izgleda "dobro". Postoji još i opcija `H` koja daje naredbu da slika mora biti



Slika 18. *Dhalia*, Janine Joles, <https://unsplash.com/photos/f0heeiu-Ec0>

točno u tekstu gdje je i napisan kod. Zatim se definiralo centriranje slike i opisa na sredinu s naredbom `\centering`, pozvala se slika te joj se dala visina 6 cm. U opis je upisan naziv te dodana naredba `\url{}` za prikaz web veze. Važana naredba je `\captionsetup{}`, bez nje se URL veza prikazuje lijevo poravnata dok u ovom slučaju treba biti centrirana (`\centering`). Zgodan detalj je `\label` koji služi za unakrsno povezivanje teksta sa slikom. U tekstu je zadana naredba `\ref{primjer}` koja će navedenu sliku uvijek povezivati s tim dijelom teksta, dati joj odgovarajući redni broj te ako se nešto promijeni taj broj ažurirati. Ako je *label* definiran sliku je jednostavno pozvati u listi slika.

Osim `h` i `t` naredbi, postoje još `b=bottom` ili dno stranice te `p=page` ili nova stranica. Ako nije nužno da slika ostane točno gdje je pozicionirana u kodu najbolje je da prepustiti  $\LaTeX$ -u da pronađe najbolju poziciju kako bi izbjegli neželjene greške i rezultate. Ako je potrebno tekst se može obaviti oko slike uz pomoć `\begin{wrapfigure}`.

$\LaTeX$  podržava formate slika `.jpg`, `.png`, `.pdf` i indirektno `.eps`. DVI verzija podržava isključivo `.eps` format.  $\LaTeX$  je osjetljiv na razmake i specijalne znakove pa ih je važno izbjegavati kod imenovanja slika.

### A.7.1. Pisanje formula i matematički mod

S obzirom na to da je  $\LaTeX$  prvotno osmišljen upravo zbog kompleksnih formula važno je spomenuti matematičke izraze unutar.  $\LaTeX$  omogućuje dva načina pisanja matema-

tičkih izraze: *inline* i *display* način. *Inline* način koristi se za pisanje formula kao dio rečenice, teksta.

Ovo je obična formula `\(x^2+y^2=z^2\)` napisana u inline modu. Može  
 ↪ se pisati i ovako `$E=mc^2$` ili ovako `\begin{math} x^n + y^n =`  
 ↪ `z^n \end{math}`.

Ovo je obična formula  $x^2 + y^2 = z^2$  napisana u inline modu. Može se pisati i ovako  
 $E = mc^2$  ili ovako  $x^n + y^n = z^n$ .

Za *display* mod koriste se

`$$formula$$`, `\[formula\]` i `\begin{displaymath} formula \end{displaymath}`

Ovo je primjer za display mod. Formula br. 1; `$$E=mc^2$$`  
 formula br.2; `\[E=mc^2\]`  
 i formula br.3: `\begin{displaymath} E=mc^2 \end{displaymath}`

Ovo je primjer za display mod. Formula br. 1;

$$E = mc^2$$

formula br.2;

$$E = mc^2$$

i formula br.3:

$$E = mc^2$$

Također se može koristiti `equation` okruženje. Poziva se naredbom `\begin{equation}`.  
 Formule dolaze s automatskom numeracijom, za isključenje treba upisati  
`\begin{equation*}`.

Primjer takvog pisanja je formula 3.

Formula s numeriranjem:

```
\begin{equation}\label{f1}
```

$$p(x) = 3x^6 + 14x^5y + 590x^4y^2 + 19x^3y^3 - 12x^2y^4 - 12xy^5 + 2y^6 - a^3b^3$$

```
\end{equation}
```

Formula bez numeriranja

```
\begin{equation*}
```

$$e^{\pi i} + 1 = 0$$

```
\end{equation*}
```

---

Formula s numeriranjem:

$$p(x) = 3x^6 + 14x^5y + 590x^4y^2 + 19x^3y^3 - 12x^2y^4 - 12xy^5 + 2y^6 - a^3b^3 \quad (3)$$

Formula bez numeriranja

$$e^{\pi i} + 1 = 0$$

Neki od simbola koji se koriste navedeni su  $\alpha\beta\gamma\rho\sigma\delta\epsilon\int\oint\sum\Pi$ , a pišu se

`\alpha` `\beta` `\gamma` `\rho` `\sigma` `\delta` `\epsilon` `\int` `\oint` `\sum` `\prod`, a pišu se

`\int` `\oint` `\sum` `\prod`. Množenje se upisuje kao `2\times2=`, potencija kao `2^n`,

razlomak kao `\left( \frac{1}{2} \right)`. Lista matematičkih formula je duga i više detalja se

može naći u knjizi *TEX: A Document Preparation System* tvorca Lesliea Lamporta.

## A.7.2. Tablice i liste

Najjednostavniji način pisanja tablica je kao ovom primjeru:

```

\begin{center}
\begin{tabular}{c c c }
\hline
stu1 & stu2 & stu3 \\
stu4 & stu5 & stu6 \\
stu7 & stu8 & stu9 \\
\hline
\end{tabular}
\end{center}

```

što rezultira ovome:

stu1	stu2	stu3
stu4	stu5	stu6
stu7	stu8	stu9

Argumenti `{c|c|c}` zapravo označavaju da će u sva tri stupca tekst biti centrirana te da će između sebe imati vertikalnu liniju. Naredba `\hline` označava horizontalnu liniju, `stu` označava tekst u stupcu a `&` separator stupaca. Kraj reda označava se s `\`. Redovi i stupci mogu se kombinirati u većoj ćeliji. Primjer ispod je primjer naredbe `\multicolumn` za kombiniranje stupaca.

Izrada listi vrlo je jednostavna, neuređena lista poziva se `\begin{itemize}` a uređena s `\begin{enumerate}`. Liste mogu biti unutar druge liste kao na primjeru:



```

\begin{enumerate}
  \item First level item
  \item First level item
  \begin{enumerate}
    \item Second level item
    \item Second level item
    \begin{enumerate}
      \item Third level item
      \item Third level item
      \begin{enumerate}
        \item Fourth level item
        \item Fourth level item
      \end{enumerate}
    \end{enumerate}
  \end{enumerate}
\end{enumerate}
\end{enumerate}

```

A rezultat je ovaj:

1. First level item
2. First level item
  - (a) Second level item
  - (b) Second level item
    - i. Third level item
    - ii. Third level item
      - A. Fourth level item
      - B. Fourth level item

### A.7.3. Citiranje i literatura

Ubacivanje literature i citiranje može se učiniti na način kako je opisanu u poglavlju 3.4 na stranici 32. Drugi način jest da se koristi odvojen dokument koristeći BibTeX. Jednom kada se takav dokument uvede u glavni dokument literatura se citira na uobičajeni način `\cite`. Takvi BibTeX dokumenti imaju ekstenziju `.bib` i uvode se u dokument pomoću `\bibliography{naziv}`, ekstenziju nije potrebno upisivati. U `.bib` dokumentu literatura se navodi nešto drugačije:

```
@article{einstein,
  author = "Albert Einstein",
  title = "{Zur Elektrodynamik bewegter K{\\"o}rper}.({German})
  [{On} the electrodynamics of moving bodies]",
  journal = "Annalen der Physik",
  volume = "322",
  number = "10",
  pages = "891--921",
  year = "1905",
  DOI = "http://dx.doi.org/10.1002/andp.19053221004"
}

@book{latex,
  author = "Leslie Lamport",
  title = "\LaTeX\ : A Document Preparation System",
  year = "1975",
  publisher = "Addison-Wesley",
  address = "Reading, Massachusetts"
}

@misc{knuthwebsite,
  author = "Donald Knuth",
```

```
title      = "Knuth: Computers and Typesetting",  
url = "http://www-faculty.stanford.edu/~{ }uno/abcde.\\html"  
}
```

Također valja biti oprezan s .bib dokumentom jer ne podržava UTF-8 znakove. Bibliografiju je najlakše povezati sa sadržajem uz pomoć tocbibind paketa kako je opisano na stranici 32. ovog rada.