

TEX

Lenart Bučar

5. marec 2017



## **Povzetek**



## Abstract



# Kazalo

<b>1</b>	<b>Uvod</b>	<b>1</b>
<b>2</b>	<b>T<sub>E</sub>X</b>	<b>3</b>
<b>3</b>	<b>Izpeljave T<sub>E</sub>Xa</b>	<b>5</b>
<b>4</b>	<b>L<sup>A</sup>T<sub>E</sub>X</b>	<b>7</b>
4.1	Matematično okolje . . . . .	7
4.2	Okolja . . . . .	7
4.2.1	Naštevanje . . . . .	7
4.2.2	Razpredelnice . . . . .	8
4.3	Paketi . . . . .	8
4.3.1	Beamer . . . . .	8
4.4	Oblikovanje besedila . . . . .	8
4.4.1	Upoštevanje prelomov vrstic . . . . .	8
4.4.2	Oblikovanje besedila v oblike . . . . .	9
4.5	Navajanje virov . . . . .	9





# Poglavje 1

## Uvod

T<sub>E</sub>X je odprtokodno programsko okolje, ki je namenjeno pisanju besedila z matematičnimi enačbami, in omogoča pisanje z t. i. makri, ki so žapakirani”v obliki paketov, med katerimi je najznamenitejši L<sup>A</sup>T<sub>E</sub>X.

Glavni razvijalec T<sub>E</sub>Xa je ameriški računalničar ter programer, Donald Knuth, prvo različico programa pa je izdal leta 1978. Knuth se je odločil za razvijanje T<sub>E</sub>Xa zaradi pisanja svoje knjige *The Art of Computer Programming*. Program je bil na začetku napisan v programskem jeziku SAIL. Kasneje je Knuth prišel na zamisel, da bi pisal tako program kot dokumentacijo naenkrat. Jezik ki ga je uporabil se imenuje WEB, končni program pa je v Pascalu.

Ena izmed zelo značilnih stvari pri T<sub>E</sub>Xu je označevanje verzij. Od verzije 3 dobi vsaka posodobitev na koncu dodatno decimalno števko, tako da se vse bolj približuje številu  $\pi$ . Trenutno smo pri verziji 3.14159265, ki je izšla leta 2014.



## Poglavje 2

# T<sub>E</sub>X

T<sub>E</sub>X temelji na podlagi ukazov, ki se začnejo z znakom `\`. Ti ukazi (macro-ji) se razširijo v besedilo ali pa v druge ukaze. Poleg tega pa lahko uporabljamo še zanke ter če-potem-drugače (if-then-else) kontrolne strukture. Ukazi imajo lahko tudi določeno število neobveznih argumentov, ki se jih po klicu ukaza našteje znotraj oglatih oklepajev (`[ in ]`), obvezne argumente pa se našteje znotraj zavitih oklepajev (`{ in }`). S temi ukazi potem spreminjamo velikost besedila, obliko, font...

Ker je T<sub>E</sub>X napisan tako dobro, teče skoraj na vseh operacijskih sistemih. Poleg tega pa Knuth ponuja denarno nagrado vsakemu, ki bi najdel hrošča (bug) v njem. Ta nagrada je na začetku znašale le 2.56 ameriških dolarjev, nato pa se je vsako leto podvojila, dokler se ni ustavila na trenutni vrednosti, ki znaša 372,6 dolarjev. Seveda pa nihče, ki je prejel to nagrado, ni čeka nikoli unovčil temveč ga je raje uokviril.



## Poglavje 3

# Izpeljave T<sub>E</sub>Xa

Na T<sub>E</sub>Xu temelji kar nekaj drugih paketov makrov, med katerimi so najznamenitejši:

- L<sup>A</sup>T<sub>E</sub>X (Lamport T<sub>E</sub>X),
- ConT<sub>E</sub>X<sub>T</sub>,
- AMS-T<sub>E</sub>X,
- JadeT<sub>E</sub>X,
- PicT<sub>E</sub>X,
- T<sub>E</sub>Xinfo in drugi.

Obstaja tudi kar nekaj razširitev, kot so BibT<sub>E</sub>X za bibliografske podatke (kot je navajanje virov), PDFT<sub>E</sub>X in drugi. T<sub>E</sub>X in vse njegove razširitve so zastonj na voljo v CTANu. CTAN ali *Comprehensive T<sub>E</sub>X Archive Network* (Popoln arhiv T<sub>E</sub>Xa) je omrežje, kamor uporabniki nalagajo svoje razširitve za T<sub>E</sub>X. Trenutno ima okoli 5500 paketov, ki jih je naložilo približno 2500 uporabnikov. Ker je program odprtokoden, je ne samo mogoče, temveč tudi zaželeno, da ga ostali uporabniki predelajo ali celo izboljšajo, edini pogoj je, da se ga ne sme distribuirati pod imenom T<sub>E</sub>X.



# Poglavje 4

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

L<sup>A</sup>T<sub>E</sub>X je ena izmed najznamenitejših distribucij makrov T<sub>E</sub>Xa, namenjena je pa pisanju knjig, znanstvenih člankov...

### 4.1 Matematično okolje

Tako kot v T<sub>E</sub>Xu je tudi v L<sup>A</sup>T<sub>E</sub>Xu zelo razvit matematični način vnašanja, ki ga med drugim uporablja tudi Wikipedia. Inicializira se ga ali z znakom \$ (T<sub>E</sub>X, deluje tudi v L<sup>A</sup>T<sub>E</sub>Xu) ali pa z \ (Samo L<sup>A</sup>T<sub>E</sub>X). V prvem primeru se ga na enak način tudi konča, v drugem pa se ga konča z \. Matematične formule lahko zapišemo v vrstici (inline, inicializira se z \$ ali \ ( ) ali pa izven vrstice (inicializira se z \$\$ ali \[ ). Primer zapisa znotraj vrstice v tem načinu je  $\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ , primer izven nje pa:

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

V matematičnem načinu lahko tudi z lahkoto zapisemo grške crke kot so  $\pi$  ( $\Pi$ ),  $\phi$  ( $\Phi$ )...

### 4.2 Okolja

Tako T<sub>E</sub>X kot L<sup>A</sup>T<sub>E</sub>X nam ponujata več različnih okolij, ki spremenijo marsikatero pravilo pisanja in s tem tudi končni izgled. Enega izmed njih smo že spoznali, to je matematično okolje, ki pa se mi zdi tako pomembno, da sem ga obravnaval posebej. Vsa okolja se začne z ukazom `\begin{ime okolja}`, konča pa z `\end{ime okolja}`.

#### 4.2.1 Naštevanje

Včasih želimo narediti seznam, in za to imamo na voljo tri okolja. Vsako se seveda obnaša malce drugače. Prvo je "itemize", pri katerem so stvari našteje z določenim simbolom pred njimi (privzeto je to pika), izgleda pa takole:

- Lorem
- Ipsum
- Dolor

Naslednje je "enumerate", ki zaporedno oštevilči predmete:

1. Sit
2. Amet
3. Consectetur

Zadnje okolje pa je "description", pri katerem imamo ime predmeta ter njegov opis:

**Adipiscing:** Elit

**Integer:** Tempor

**Est:** Ut Ux

### 4.2.2 Razpredelnice

Zelo pogosto, še posebej v strokovnih člankih, poročilih itd. moramo dodati kakšno razpredelnico.  $\text{\LaTeX}$  nam to omogoča z uporabo okolja "tabular"

## 4.3 Paketi

Seveda se lahko zgodi, da si pisec zaželi nek ukaz, ki ga  $\text{\LaTeX}$  sam po sebi ne ponuja. Sedaj ima na voljo dve možnosti. Lahko sam s pomočjo že obstoječih ukazov napiše novega, lahko pa pogleda ali je to že kdo naredil. Ker se ponavadi izkaže da je, lahko to kodo doda v svoj program z uporabo ukaza `\usepackage`. Glede na razširjenost programa, niti ni presenetljivo, da obstajajo paketi za skoraj vse. Od takih, ki omogočajo lažje pisanje znanstvenih enot, pa do takega, pri katerem s pomočjo enega samega ukaza dodamo "Lorem Ipsum" besedilo za zapolnjevanje prostora.

### 4.3.1 Beamer

$\text{\LaTeX}$  ponuja tudi možnost izdelave predstavitve, z uporabo paketa beamer. Pri tem paketu, uporabnik definira kako naj izgleda posamezna prosojnica, kateri elementi naj se kdaj prikazujejo...

## 4.4 Oblikovanje besedila

$\text{\LaTeX}$  pri generiranju končnega besedila odstrani vse večkratne presledke, prehode v novo vrstico itd. Zato sta si besedili: "Lorem Ipsum" in "Lorem Ipsum" enaki. Po končanem generiranju namreč pri prvem dobimo Lorem Ipsum, pri drugem pa Lorem Ipsum. Če na primer vpišemo

```
L
o
r
e
m
I
p
s
u
m
```

se nam izpiše L o r e m I p s u m.

Lahko pa se nam zgodi, da želimo besedilo oblikovati točno na določen način.

### 4.4.1 Upoštevanje prelomov vrstic

Če želimo, da  $\text{\LaTeX}$  upošteva prelome vrstic, lahko uporabimo ukaz `\obeylines`. Če torej recimo napišemo:

```
{\obeylines
Lorem Ipsum
Dolor Sit
Amet
}
```



Se nam izpiše:

Lorem Ipsum  
Dolor Sit  
Amet

Če pa tega ukaza ne bi uporabili, bi z istim vhodom dobili: Lorem Ipsum Dolor Sit Amet.

#### 4.4.2 Oblikovanje besedila v oblike

### 4.5 Navajanje virov

L<sup>A</sup>T<sub>E</sub>X nam z pomočjo dodatka BibT<sub>E</sub>X omogoča preprosto navajanje virov. Potrebujemo le dodatno *.bib* datoteko, ki jo na zelenem mestu kličemo z `\bibliography{ime datoteke}`. Datoteka mora biti seveda pravilno oblikovana, da jo L<sup>A</sup>T<sub>E</sub>X lahko prebere. Vsako delo, mora imeti svoj “odstavek”, ki pa izgleda takole:

```
@tip dela { interna oznaka, ki jo bomo uporabljali za citiranje,
  author = {},
  title = "",
  journal = "",
  volume = "",
  number = "",
  pages = "",
  year = "",
  DOI = ""
  publisher = "",
  year = "",
  url = "",
  note = "",
}
```