

A Latex használata (3. rész)

Felsorolások, táblázatok és ábrák

A sorozatnak ebben a részében a szövegformázás további eszközeiről lesz szó. Néhány példát mutatok felsorolások, táblázatok és ábrák beillesztésére. Az ábrákkal kapcsolatban kitérek olyan rajzolóprogramokra, melyek képesek hatékonyan együttműködni LaTeX-hel. Most sem célok a technikai részletek alapos tárgyalása.

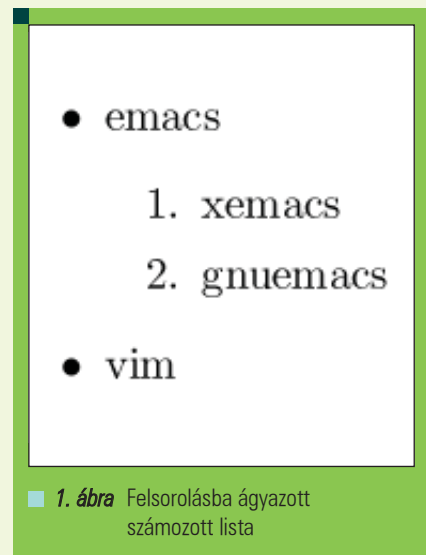
Latex-ben többfajta felsorolás van, számozott vagy számozás nélküli és leíró. Számozás nélküli listát a `\begin{itemize} \end{itemize}` környezettel írhatunk, ahol az egyes elemeket az `\item` jelzi. Számozott listák esetén az `enumerate` helyett az `enumerate` környezetet alkalmazzuk, az elemeket itt is az `\item` jelöli, ennek most van opcionális paramétere, amivel az elem előtt álló számot állíthatjuk be. Leíró listákat a `description` környezettel készíthetünk. Itt megadunk egy címszót, amit félkövéren szed a LaTeX, és utána a leírását. Ekkor az `\item` opcionális paraméterében van az a kifejezés, aminek megadjuk a leírását. Emacs-et és auctex-et használva a LaTeX menüpont Insert Environment alpontjában az előbbi környezeteket választva szúrhatunk be felsorolásokat a forrás fájlba. Leíró listák esetén az auctex a minibufferben, ami az emacs ablak legalsó sora, rákérdez az `\item` paraméterére. A különböző típusú felsorolások négy szintig egymásba ágyazhatók. Az elemek előtt álló szimbólumok vagy számok megváltoztathatók. A `\renewcommand{}{}{}` már definiált LaTeX parancsok átdefiníálására való, az első paramétere az a parancs amit megváltoztatunk, a második az új definíció.

Hatása a következő újradefiniálásig tart. Számozott listáknál az első szinten arab számok az alapértelmezettek, további szinteken római számok és betűk lesznek. Az első szinten a következő paranccsal állíthatjuk be számozás stílusát: `\renewcommand{\labelenumi}{\Alph{enumi}}` További szinteken a `\labelenumii`, `\labelenumiii` és a `\labelenumiv` parancsokat kell átdefinálni.

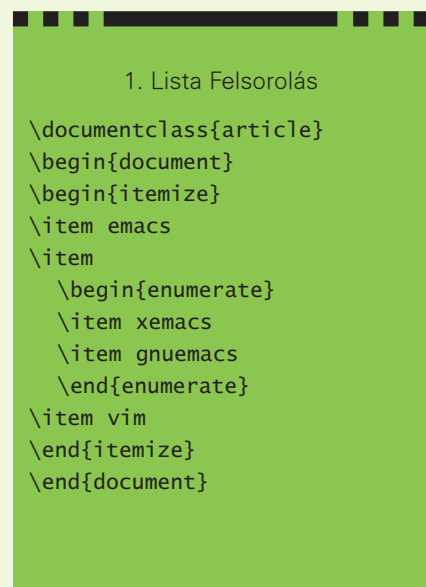
Lehetséges értékeik:

- `\arabic` arab számok
- `\alph` kis latin betűk
- `\Alph` nagy latin betűk
- `\roman` kis római számok
- `\Roman` nagy római számok

A felsoroló listáknál használt szimbólum is átállítható a `\renewcommand{\labelitemi}{\star}` paranccsal, ahol a használható szimbólumok a `symbols.dvi` fájlban találhatóak. Debian alatt ez a fájl a `/usr/share/doc/texmf/latex/general` könyvtárban van. Előbbihez hasonlóan, a további szinteken a `\labelitemii`, `\labelitemiii` és a `\labelitemiv` definíálja szimbólumokat.



1. ábra Felsorolásba ágyazott számozott lista



2. Lista Táblázat

```
\documentclass{article}
\begin{document}
\begin{tabular}{|1|1|1|}
\hline
1 & 4 & 7 \\ \hline
2 & 5 & 8 \\ \hline
3 & 6 & 9 \\ \hline
\end{tabular}
\end{document}
```

1	4	7
2	5	8
3	6	9

■ 2. ábra Háromszor hármast, teljesen bekeretezett táblázat, amiben az oszlopok balra igazítottak

3. Lista PostScript fájl beillesztése

```
\documentclass[a4paper,12pt]
\begin{document}
\usepackage{graphics}
\resizebox{70mm}{1}{
\rotatebox{90}{
\includegraphics{3_
Balasz.eps}
}
}
\end{document}
```

Táblázatok közül csak a leg-egyszerűbbekkel foglalkozom. Ugyan szinte tetszőleges bonyolultságú táblázat elkészíthető *LaTeX*-ben, de ehhez nagy tapasztalat és több, egymással részben inkompatibilis csomag betöltése szükséges. Fontos, hogy a táblázatok geometriai jellemzőivel nem kell törődnünk, azokat a *LaTeX* meghatározza. Lehetőségünk van ezeket is megadni, amivel azonban csak különleges esetekben érdemes élni.

A *LaTeX* táblázatok

a `\begin{tabular}` `\end{tabular}` környezetben található. A kötelező paraméterben adjuk meg az oszlopok számát és igazítását. Itt az oszlopokat az `lcr` betűk valamelyike jelzi, utalva az igazításra, `l`: balra, `c`: középre, `r`: jobbra. Az oszlopok között függőleges vonalakat húzhatunk a `|`-al. A táblázat sorait `\\` választja el, míg a soron

belül a cellákat az `&`. Vízszintes vonalakat `\hline`-al illeszthetünk a táblázatba.

Az esetleges üres cellákat is jelezni kell, az oszlopok számánál eggyel kevesebb `&` jelnek kell lennie a táblázat soraiban.

Ezek egyszerű táblázatok esetén ez megegyezhetnek az *emacs* soraival, bonyolultabb táblázatoknál ez általában nincs így.

Ábrákat több, lényegesen különböző módon illeszthetünk *LaTeX* dokumentumainkba. A csak vonalakkal, körívekkel álló ábrákat elkészíthetjük *LaTeX* utasításokkal, ez azonban elég nehézkes. Másik lehetőség, hogy rajzolóprogramot használunk és az állítja elő a *LaTeX* kódot. Ilyen a *kig* vagy az *xfig*.

A *gnuplot* is hatékonyan használható *LaTeX* ábrák előállítására, különösen formulával megadott függvények vagy mérési adatok, grafikonok esetén. *Gnuplot*-ban élesen kettéválik az ábrák elkészítése és megjelenítése. Előbb szöveges módban, nem interaktívan leírjuk a grafikont majd külön lépésben kirajzoltatjuk képernyőre vagy fájlba. Az első lépést segíti az *emacs gnuplot* módja.

A *LaTeX* kód kézzel vagy programmal történő előállításán kívül egy további lehetőség *postscript* fájlok beillesztése. Ehhez a *graphics* csomagot kell betölteni. Ezután a következő kódrészlettel szűrhatunk be *eps* fájlokat.

Belülről kifeje fogom elemezni a fenti kódot. Ebben szerepel a *TeX* egyik belső, felhasználók elől általában elrejtett fogalma, a doboz. Segítségével meglehetősen bonyolult

formázásokat elérhetünk, amit részletesebben nem vizsgálók.

Az `\includegraphics{}` utasítással adjuk meg az *eps* fájl nevét.

A `\rotatebox{ }{ }` elforgatja az első paraméterben adott fokkal a másodikban szereplő dobozt, ami esetünkben a *postscript* fájl.

A `\resizebox{ }{ }{ }` az első két paraméternek megfelelően átméretezi a harmadikban lévő dobozt. Az elsőben a vízszintes, a másodikban függőleges méret van. A `!` hatására a paraméter értékét a másik, számokkal megadott alapján úgy választja meg a *LaTeX*, hogy az ábra köré rajzolt téglalap oldalainak aránya ne változzon. Az ábrák beillesztése szerintem azon ritka esetek egyike, amikor érdemes kézzel megadni a geometriai méreteket mm-ben. *PostScript* ábrák használatakor a `pdf\atex` nem működik, a *PDF* formátumot több lépésben kell előállítani. A konvertálás a `dvi/pdf`-el végezhető el, ami nem érhető el *emacs*-ből.

Mint általában a *TeX*-ben, felsorolások, táblázatok vagy grafikák esetén is, a gyakran előforduló, egyszerű feladatok könnyen megoldhatók. Azonban különösen jellemző hogy a bonyolultabb megoldásához több, egymást részben kizáró, *LaTeX* csomagot és külső programokat kell használni. Valamint, hogy nem elkerülhető a *LaTeX* belső felépítésének megismerése. Fontos, azonban, hogy már a *LaTeX* által biztosított eszközök alapértelmezett beállításával is általában elérhetjük céljainkat, esetleg kis kompromisszumok árán.

A sorozat további következő részében a hosszú, több szerzős dokumentumok szerkesztéséről lesz szó.

Ehhez a *LaTeX* és az *emacs* kényelmes eszközöket kínál. Nyomon követhető a változtatások, visszatérhetünk régi változatokhoz, azokból részleteket emelhetünk át.



Vecsei Balázs

(vecseib@math.bme.hu)

A BME TTK-n végzett matematikusként.

1998 óta foglalkozik

Linuxszal és *LaTeX*-hel.

Szabadidejében sokat túrázik Magyarországon és külföldön is, kedvence a Börzsöny.