

Debreceni Egyetem

Informatikai Kar

Tipográfiai alapismeretek és  $\text{\LaTeX}$  2 $\epsilon$   
alapú szövegszerkesztés tanítása  
középiskolában

Szakdolgozat

*Témavezető:*

**Dr. Bujdosó Gyöngyi**

egyetemi adjunktus

*Készítette:*

**Szabó Csaba**

informatika tanár szakos

hallgató

Debrecen

2008.

# Köszönetnyilvánítás

Köszönöm témavezetőmnek, Dr. Bujdosó Gyöngyinek azt a minden apró részletre kiterjedő gondos figyelmet, amellyel a dolgozat megszületését mindvégig kísérte. Számtalan hasznos tanácsa komoly segítséget jelentett.

Köszönettel tartozom továbbá családomnak azért a megértő türelemért, amellyel a szakdolgozat írásának időszakában körülvett.

# Tartalomjegyzék

<b>Előszó</b>	<b>6</b>
<b>1. Bevezetés</b>	<b>8</b>
1.1. Tipográfia és szövegértés . . . . .	8
1.2. A L <sup>A</sup> T <sub>E</sub> X és az emelt szintű érettségi . . . . .	9
1.3. A szakdolgozat felépítése . . . . .	10
<b>2. Tipográfiai alapismeretek</b>	<b>11</b>
2.1. A tipográfia fogalma . . . . .	11
2.2. A tipográfia célja . . . . .	12
2.3. Betűk, betűváltozatok, betűcsaládok . . . . .	13
2.3.1. Betűk . . . . .	13
2.3.2. Ligatúrák . . . . .	14
2.3.3. Betűváltozatok . . . . .	15
2.3.4. Betűméret . . . . .	16
2.4. Írásjelek . . . . .	17
2.4.1. Mondatvégi írásjelek . . . . .	17
2.4.2. Központosítás, idézetek . . . . .	17
2.4.3. Kis vízszintes vonalak . . . . .	18
2.5. Üres helyek . . . . .	19
2.5.1. Betűköz . . . . .	19
2.5.2. Szóköz . . . . .	20
2.5.3. Sorköz . . . . .	21
2.6. Szövegközi kiemelés . . . . .	21
2.6.1. Nem javasolt módozatok szövegközi kiemelésre . . . . .	22

---

2.6.2.	Javasolt módozatok szövegekőzi kiemelésre . . . . .	23
2.7.	Tördelés . . . . .	23
2.7.1.	Margók, szedéstükör . . . . .	23
2.7.2.	Címrendszer . . . . .	24
2.8.	Olvashatóság . . . . .	26
<b>3.</b>	<b>Szövegszerkesztés <math>\LaTeX</math> környezetben</b>	<b>28</b>
3.1.	Alapok . . . . .	28
3.1.1.	A $\TeX$ és a $\LaTeX$ . . . . .	29
3.1.2.	A $\LaTeX$ előnyei és hátrányai . . . . .	29
3.1.3.	Egy $\TeX$ rendszer telepítése . . . . .	31
3.1.4.	Egy $\LaTeX$ forrásállomány felépítése . . . . .	33
3.1.5.	Begépelés, fordítás, kimeneti fájl . . . . .	34
3.2.	A kiadvány tipográfiájának beállítása . . . . .	36
3.2.1.	Dokumentumosztályok . . . . .	36
3.2.2.	Oldalformátum . . . . .	37
3.2.3.	Betűk . . . . .	38
3.3.	A szöveg szedése . . . . .	39
3.3.1.	Környezetek . . . . .	39
3.3.2.	Igazítás . . . . .	39
3.3.3.	Sortörés, oldaltörés . . . . .	40
3.3.4.	Címrendszer, címhierarchia . . . . .	41
3.3.5.	Szóközök, térközök . . . . .	42
3.3.6.	Felsorolás, számozás . . . . .	43
3.3.7.	Képek és táblázatok . . . . .	43
3.3.8.	Idézetek, versek . . . . .	46
3.4.	Matematika, képletek . . . . .	47
3.4.1.	Definíciók, tételek, bizonyítások . . . . .	47
3.4.2.	Matematikai formulák leírása . . . . .	48
3.5.	Jegyzetek, jegyzékek . . . . .	51
3.5.1.	Tartalomjegyzék létrehozása . . . . .	51
3.5.2.	Lábjegyzet, széljegyzet, végjegyzet . . . . .	52
3.5.3.	Irodalomjegyzék beillesztése . . . . .	52

3.6. A dokumentum elektronikus megjelenítése . . . . .	53
3.6.1. Prezentációkészítés PDF-alapokon . . . . .	53
<b>Összegzés</b>	<b>56</b>
<b>Irodalomjegyzék</b>	<b>60</b>

# Előszó

Szakdolgozatom témaválasztását egyebek mellett az a törekvésem motiválta, hogy középiskolai tanárként egyszerre tudjak használható tudást átadni és általános műveltséget közvetíteni egy adott – viszonylag szűk és speciális, de általam fontosnak vélt – témakörben.

A magyar közoktatásban jelenleg érvényben lévő kerettantervnek az informatikai műveltségterületre vonatkozó fejezetei meghatározzák, hogy szövegszerkesztési ismereteket a középiskola kilencedik évfolyamán kell elsajátítaniuk a tanulóknak. „Dokumentumkészítés számítógéppel · Többféle formázási megoldást tartalmazó dokumentumok készítése. Típusdokumentumok elkészítése minta alapján. Kiselőadás, esszé vagy tanulmány dokumentumának elkészítése (képekkel, táblázatokkal, diagramokkal, például környezetvédelmi vagy informatikai témáról).” [16] (Maglehetősen sajnálatos egyébiránt, hogy a kötelező órakeret későbbi évfolyamokon egyáltalán nem tartalmaz számítástechnikai, informatikai tantárgyat.)

Tapasztalataim szerint a „szövegszerkesztési” ismeretek tanítása az esetek döntő többségében nélkülözi a tipográfiai megalapozást és jobbára pusztán a szövegszerkesztő program használatának elsajátíttatására korlátozódik. Magától értetődő módon az iménti megállapítás nem értékítélet. A rendelkezésre álló időkeret a kötelező informatika órákon nem is tesz lehetővé ennél többet. Mindazonáltal fontosnak gondolom, hogy az általunk, középiskolai tanárok által érettségi vizsgákra és egyetemi-főiskolai továbbtanulásra felkészített diákok általános műveltségének részét képezze az is, hogy hogyan kell megjelennie egy általuk készített dokumentumnak akkor, ha azt az idők során írott és íratlan szabályokká erősödő tipográfiai hagyományoknak megfelelően akarják elkészíteni.

Nem elhanyagolható szempont a tanulók szövegértésének elősegítése sem, a kiadványok olvashatóságának javításával. Az iskolás korú gyermekek szövegértési képességeit (is) vizsgáló PISA-felmérésekben a magyar diákok rendre a mezőny sereghajtói között végeztek európai összevetésben. Nehezen volna vitatható, hogy a közoktatásnak jócskán van tennivalója e téren. Az egyik lehetőség például a tanulók kezébe kerülő dokumentumok olvashatóságán javítani a tipográfia széleskörű eszköztárát igénybe véve. Vonatkozik ez a tanári oldalra is: kívánatos volna a középiskolában oktatók – amúgy technikailag, szoftverkezelésileg jónak mondható – szövegszerkesztési ismereteit „feljavítani”. A legfontosabb tipográfiai szabályok, szempontok egyszerűek, de széles körben nem ismertek. A jelen szakdolgozat egyik fejezetét képező tipográfiai összefoglaló tehát szaktól függetlenül valamennyi kollégámnak is szól.

A tananyagban szereplő szövegszerkesztési ismeretanyag elsajátításához elegendő valamely WYSIWIG-típusú program használatának megtanítása. Ha azonban ezen túlmutató célokat fogalmazunk meg, és nyomdai minőségű, igényes tipográfiájú kiadványok készítését akarjuk megtanítani, akkor a L<sup>A</sup>T<sub>E</sub>X szövegszedő rendszer tűnik megfelelő választásnak. Egy másik ok, ami mellette szól, az az, hogy a L<sup>A</sup>T<sub>E</sub>X-et egyfajta programnyelvnek fogva fel, a programozói szemléletű gondolkodásra való „ránevelésnek” is remek eszköze lehet. Utóbbi fontos és hasznos kiegészítése volna az emelt szintű informatika érettségire való felkészítésnek (ld. az 1.2. szakaszt).

A kerettantervi követelmények alapján egyértelműnek tűnik, hogy a tipográfiai ismeretek és a L<sup>A</sup>T<sub>E</sub>X-alapú szövegszerkesztés oktatása kívül kell hogy essék az informatikára jutó kötelező órakereten, és elsősorban fakultatív tárgy formájában, szakkörként valósítható meg.

Jelen szakdolgozatban arra teszek kísérletet, hogy egy ilyen típusú és tematikájú szakkörhöz oktatási segédanyagot állítsak elő. Egy tartalmi összefoglalót, amely alapvetően a szakkört tartó tanárnak készül, de amelyet reménység szerint szakköri füzetként is haszonnal olvashatunk majd. A tananyag-feldolgozást tanári oldalról kidolgozott óravázlatokkal próbálom meg segíteni.

# 1. fejezet

## Bevezetés

### 1.1. Tipográfia és szövegértés

Azt a hipotézist, hogy van összefüggés a szövegértés és a tipográfia között, kísérleti úton is igazolták már [2]. 1218 magyar iskolás gyermeket vontak be abba a felmérésbe, amelynek során azt vizsgálták, hogy a szöveg tipográfiája (a betű típusa, mérete, formája vastagsága) befolyásolja-e az olvasottak pontos értését. Különböző korú általános iskolásoknak különböző szövegeket választottak ki, majd ezekből négyféle nyomtatott formát állítottak elő. Volt nagyobb (14 és 12 pontos) és kisebb (11 pontos) betűmérettel szedett nyomtatvány. Utóbbiból három verzió is készült: az egyikben csak álló, a másikban csak dőlt betűváltozat fordult elő, míg a harmadik álló, dőlt és félkövér betűket egyaránt tartalmazott.

Az átlagosan tizenöt percnyi olvasási idő leteltével a résztvevőknek ellenőrző kérdésekre kellett válaszolniuk, írásban, időkorlát nélkül. Az olvasási időtartam függetlennek bizonyult a szöveg tipográfiájától, a szövegértést ellenőrző kérdésekre adott helyes válaszok száma azonban szignifikáns módon korrelált azzal: a leggyengébb eredmények valamennyi korcsoportban a dőlt betűs szövegek esetében születtek, a legjobbak pedig a „vegyes” szedés esetén. A kísérletet végzők ezt a RANSCHBURG-féle homogén gátlás csökkenésével, illetve megszűnésével magyarázták.

A felmérés végső konklúziója az volt, hogy az életkor növekedésével csök-

ken ugyan a tipográfia teljesítménybefolyásoló hatása, de a vizuális bemenet mindvégig meghatározó marad. Véleményem szerint ez felnőttkorban sincs másképp, bár ennek némileg ellentmond egy másik felmérés [5], amelyben a szerző egyetemisták körében vizsgálódott. Ő nem tudott szignifikáns különbséget kimutatni a jó tipográfiájú szöveget olvasó és megtanuló vizsgálati csoport és a tipográfiától úgyszólván mentes olvasnivalóval küzdő kontrollcsoport felismerési és felidézési teljesítménye között.

Kétségtelen tény ugyan, hogy ma már egy számítógép segítségével létrehozott kiadvány nem feltétlenül – sőt, egyre kevésbé – testesül meg papír alapú végtermék formájában, viszont a publikációk elektronikus, alapvetően webes megjelentetése legalább akkora formai körültekintést igényel, illetve igényelne, mintha az adott dokumentum nyomtatott formában kerülne az olvasó kezébe.

## 1.2. A L<sup>A</sup>T<sub>E</sub>X és az emelt szintű érettségi

A L<sup>A</sup>T<sub>E</sub>Xhel való megismerkedésnek a közvetlenül felhasználható ismeretanyag, tudáson túl egy közvetett haszna is megfogalmazható. A rendszert tekinthetjük egyfajta programnyelvnek, miáltal a tanulók programozói szemlélete, gondolkodása fejleszthető ezzel az eszközzel. Az emelt szintű informatika érettségi gyakorlati vizsgarészének tekintélyes hányada – egészen pontosan 60 %-a – algoritmizálási, programozási ismereteket kér számon.

A 40/2002. (V.24.) OM-rendelet [17]ebben a tárgykörben a következőket írja elő a vizsgázók számára: „Ismerje a strukturált programozás alapelveit, a lehetséges programszerkezeteket. Tudja a szükséges változókat kiválasztani és programbeli használatukat szabatosan megfogalmazni. Tudja pontosan leírni az egyes típusfeladatok kiinduló állapotát (azaz felsorolni az értékkel rendelkező változókat és tulajdonságukat) és a várt eredményt (azaz mely változóba, milyen feltételek mellett, milyen értékeket kell visszaadnia a programnak). Tudja leírni a megfelelő algoritmusokat valamely algoritmus-leíró nyelven.”

Fentiek alapján egyértelműnek tűnik, hogy a tipográfiai ismeretek és a L<sup>A</sup>T<sub>E</sub>X-alapú szövegszerkesztés oktatása kívül kell hogy essék az informati-

kára jutó kerettantervi kötelező órakereten, és elsősorban fakultatív tárgy formájában, szakkörként valósítható meg.

### 1.3. A szakdolgozat felépítése

A tartalmi összefoglaló két fő részből áll. Az első rész jobbra elméleti jellegű, ebben veszem végig azokat a tipográfiai definíciókat, szabályokat, hagyományokat, amelyek megalapozni hivatottak az alapfoknál magasabb szintű szövegszerkesztési ismeretek megtanulását. Fontos leszögezni, hogy a tipográfia szokásos felosztása alapján, mely szerint megkülönböztetünk betűtervezést és betűvel való kiadványtervezést, kizárólag az utóbbira szorítkozom, és legfeljebb ajánlott irodalom formájában utalok az előbbire. A második részben következik a gyakorlat: szövegszerkesztés L<sup>A</sup>T<sub>E</sub>X segítségével. Itt a tananyag kidolgozása szempontjából a fő feladat a résztémakörök közötti erős szelekció. A nélkülözhetetlen alapismereteken kívül (forrásfájl, preambulum, dokumentumosztályok, stb.) néhány olyan témakör beillesztésére kerülhet csak sor, amelyeket a tanulók már középiskolai pályafutásuk alatt is a gyakorlatban alkalmazhatnak pl. házi dolgozatok vagy kiselőadásukat támogató prezentációk készítése során. Gondolok itt képek, ábrák, diagramok dokumentumba illesztésére, matematikai, fizikai, kémiai képletek szedésére, illetve az elkészült dokumentumok HTML-be vagy PDF-be konvertálására, elektronikus úton történő publikálás céljából.

## 2. fejezet

# Tipográfiai alapismeretek

### 2.1. A tipográfia fogalma

Valamirevaló klasszikus tudomány tárgyalása nem kezdődhet másként, mint hogy „már az ókori görögök is...” Nos, a mi esetünkben ez nyilvánvalóan nem így van, hiszen a könyvnyomtatás Európában a XV. században kezdődött, de a görögökkel való kapcsolatot itt sem kell nélkülöznünk. A tipográfia szó ugyanis a tüposz (jelentése vert vagy vésett ábra) és a gráfó (írni) görög szavak összevonásával keletkezett és nyert új jelentést: típusokkal írni [10]. Milyen elvárásokat támasztunk egy dokumentummal szemben, azon túl, hogy tartalmilag értékes, informatív és nyelvtanilag helyes legyen? Nyilvánvalóan fontos az is, – vagy legalábbis fontosnak kellene lennie mindenkinek számára – hogy a kiadvány esztétikus megjelenésű és könnyen, jól olvasható legyen. Ennek megfelelően a tipográfia nyomtatott betűkkel foglalkozó tudomány, melynek célja olyan írásképet kialakítani betűtípusok és betűcsaládok alkalmazásával, mely egyszerre esztétikus és célszerű. Modernebb megközelítésben az információ megjelenítésének szabályrendszere [Wikipédia]. Két fő tevékenységi körre osztható. Az egyik a betűtervezés, a másik pedig a betűalkalmazás, vagy másképpen a betűvel való tervezés. Jelen dolgozatban csak az utóbbival foglalkozunk.

## 2.2. A tipográfia célja

A tipográfia alapvető céljai közül feltétlenül említést érdemelnek a következők: figyelemfelkeltés, olvashatóság javítása, fontos részek kiemelése, rendszerezés, egységes megjelenés kialakítása, ezen keresztül esztétikai élmény nyújtása.

– Figyelemfelkeltés

Ha a tipográfia módszereit következetesen alkalmazzuk, akkor kiadványunkat kézbe véve (vagy elektronikus megjelenés esetén a monitorra pillantva) már az első benyomás oly mértékben kedvező lehet, hogy a potenciális olvasó figyelmét jó eséllyel felkelti.

– Olvashatóság javítása

Minden kiadvány az olvasónak készül, következésképpen a tipográfiai kialakításnak is az olvasó kényelmét kell szolgálnia: a dokumentum jól áttekinthető, logikus szerkezetű, könnyen olvasható legyen.

– Fontos részek kiemelése

Az olvasó figyelmének a lényeges tartalmi elemekre irányítása megkerülhetetlen tipográfiai feladat. A cél elérését a szövegközi kiemelések különböző formáival valósíthatjuk meg.

– Rendszerezés

Mondanivalónk hatékony közvetítéséhez szükség van a tartalmi szerkezet formai megjelenítésére is. Ehhez grafikai, tipográfiai eszközöket használhatunk. Ilyen lehet egy jól átgondolt, logikus címrendszer kialakítása.

– Egységes megjelenés kialakítása

„A szép szerkesztés esztétikai, a szakszerű szerkesztés anyagi értéket is képvisel, melyekkel kiadványaink tartalmának súlyát fokozhatjuk” [8].

## 2.3. Betűk, betűváltozatok, betűcsaládok

A tipográfia alapeleme a betű. „Egy könyv annál mintaszerűbb, minél tisztábban érvényesül benne a betű egyszerű szépsége. Róla beszél, rajta nyugszik a könyvművészet dicsősége. És joggal, mert egyedül a betűk állnak meg szükségszerűen önmagukban, minden egyéb azonban csak rajtuk keresztül” – írta a XVIII. században máig érvényes igazságként GIAMBATTISTA BODONI. A tipográfia egyik paradoxona, hogy a betű az alapegység, ugyanakkor nem betűket, hanem szóképeket olvasunk, így a betűknek egyszerre kell egymástól jól elkülöníthetőeknek, de egyben egymáshoz könnyen kapcsolódóknak is lenniük [3]. A tipográfiában a betűkép egy összhangba hozott betűtervsorozat, ami egy teljes ábécét alkot, amelyet nyomtatásban vagy a számítógép képernyőjén meg lehet jeleníteni. [Wikipédia]

### 2.3.1. Betűk

A betűk megtervezésének folyamatával, művészetével e helyütt nem foglalkozunk, de a téma iránt érdeklődők számára egy lehetséges kiindulópont lehet a typoblog [18]. Betűtípusnak nevezzük az azonos tipográfiai elv alapján készített, azonos formajegyekkel rendelkező, általában a betűtípus alkotójának nevét viselő betűsort és a hozzá tartozó írásjeleket [12]. Dokumentumainkhoz a betűk típusát úgy érdemes és célszerű megválasztani, hogy a választott betűtípus stílusa minél jobban illeszkedjen a szöveg stílusához. Néhány javaslat a [10] alapján: szépirodalmi szövegekhez jól illenek a reneszánsz típusok, ilyen pl. a széles körben hozzáférhető és nagyon szép Garamond; társadalomtudományi cikkekhez válasszunk barokk betűt, pl. Times-t; műszaki témájú dokumentumokhoz pedig klasszicista betűtípust, mint például a Bodoni.

A betűtípusokat számos szempont szerint szokás osztályozni. Megkülönböztetünk például talpas (serif) és talp nélküli (sans serif) betűket (a sans franciául azt jelenti, hogy „nélkül”, tehát sans serif, azaz betűtalp nélküli). Utóbbinak állandó vonalvastagságú változatát *groteszk* betűtípusoknak is szokták hívni. Mindkét típusból nagy számban van választék, egyes típusok széles körű használatban vannak, mások csupán díszítőelemként fordulnak elő. A serif betűtípust könnyebb olvasni hosszabb szövegben, mint a talp

nélkülít, ezért a legtöbb könyvben és újságban ezeket használják (legalábbis a főbb cikkekben). Ellentétben a nyomtatott szöveggel, a talpatlan betűk jobban olvashatóak a számítógép képernyőjén. A leggyakrabban használt modern sans-serif betűtípus a Verdana nevezetű.

A szakkör tananyagának következő részében, a L<sup>A</sup>T<sub>E</sub>X rendszer alapelemeinek ismertetésekor látjuk majd, hogy más szempont alapján is csoportosíthatunk: a talpasság mellett a betűk állandó vagy változó vonalvastagságára is tekintettel lehetünk. A talpas és nem egyenletes vonalvastagságú betűk családjának neve *antikva*, míg a talp nélküli betű, melynek minden vonala ugyanolyan vastag, a *groteszk*.

Idegen szóval proporcionalitásnak nevezik a betűk következő fontos tulajdonságát, ez magyarul annyit tesz, hogy a betűk egyforma szélesek-e, vagy sem. A klasszikus írógépeken, vagy DOS-os terminálablakokban minden betű azonos szélességű helyet foglal el, míg a számítógéppel szerkesztett dokumentumokban leggyakrabban használt betűk változó szélességűek, azaz *proporcionálisak*.

### 2.3.2. Ligatúrák

Valószínűleg nincs olyan tipográfiával foglalkozó kiadvány, amely a betűkről szóló részében ne tenne említést az ún. ligatúrákról. Itt se tegyünk kivételt. A ligatúra, magyarul ikerbetű nyomdászati szakkifejezés. Két vagy több betű egybeírását, összekapcsolását, egyesítését jelenti, különösen régi kéziratokban és az ősnymtatványokban volt gyakori. A középkori írásgyakorlatban – körülbelül az V–VIII. században, a karoling írásreform előtt – jellemző volt, hogy folyamatos kézmozdulattal, az íróeszköz felemelése nélkül minél több betűt írtak le, még ha ez a betűk torzulásával vagy helyzetük megváltozásával járt is. Így a kódexmásolók is nagyon sok rövidítést, betűösszevonást használtak, egyrészt, hogy munkát takarítsanak meg, másrészt, hogy a sorokat megfelelően ki tudják tölteni. A XV–XVI. századi nyomdák gyakran használtak ligatúrákat: GUTENBERG 42 soros Bibliájában 290 betűjel fordult elő, ezek nagy része ligatúra volt. A ligatúrákat kétféleképpen hozták létre: a betűket egymásba tolták vagy ívvel kötötték össze. (Forrás:

Magyar Nagylexikon) A mai magyar tipográfiában az f-ligatúrák használata a leggyakoribb. Az f-ligatúrák teljes készlete az ff, fi, fl, ffi, ffl betűpárok-ból, illetve betűhármassokból áll. Alapesetben csak az első három lehető fel gépünk betűkészleteiben, ilyenkor először a különálló f-et, majd pedig a ligatúrát kell szedni.

### 2.3.3. Betűváltozatok

Egy adott betűtípus különböző változatai együtt alkotják a *betűcsaládot*. Ilyen változatokat megkülönböztetünk például a betűk formája alapján: álló, *kurzív*, KISKAPITÁLIS. A kurzívot gyakran nevezik *dőlt*nek, illetve olaszosnak (*italic*). Ez nem azonos a *döntött* (*slanted*) betűvel, melyet a magyar tipográfiában nem is szabad használni. A kiskapitális pedig annyit tesz, hogy a kisbetűk helyén és méretében nagybetűk szerepelnek. Ezt a változatot a betűtervezők nagyon gondosan megtervezik, hogy harmonikusan illeszkedjen a kisbetűk közé. Arányban a rendes nagybetűkhöz képest szélesebbek, fővonaluk pedig vastagabbak. Ebből természetesen az is következik, hogy a „sima” nagybetűk mechanikus lekicsinyítése egyáltalán nem eredményez kiskapitális betűt! A számítógépes szövegszerkesztő programok mégis viszonylag gyakran nyúlnak a kicsinyítés nemtelen eszközéhez, elcsúfítva ezzel a dokumentum kinézetét.

Feltétlenül törekedni kell arra, hogy egy-egy betűtípushoz valamennyi változatot beszerezzük, mert a szöveg képe csak így lehet harmonikus.

A szövegek többségénél nem oldható meg a tipográfiai feladat egyetlen betűtípus egyetlen változatának használatával. (Egyetlen betűtípussal viszont igen! Egy dokumentumon belül csak súlyosan indokolt esetben használjunk többet!) Ha nem alkalmazzuk alkalmas helyeken a kurzív, a félkövér és a kiskapitális változatokat, akkor a mégoly érdekesítő szöveg is könnyen unalmassá válhat, másrészt pedig a leírtak megértése lesz lényegesen nehezebb. A betűváltozatok használatáról még lesz szó a szövegek közti kiemelésről szóló részben.

### 2.3.4. Betűméret

Már a nyomdai tipográfia történetének kezdetén világos volt a nyomdászok és betűöntők számára, hogy nagyon apró fokozatokat kell megkülönböztetni, ezért a milliméternél sokkal finomabb egységre volt szükség. (Hogy egyáltalán szükség volt egységre, az nyilvánvaló.) Így jött létre a nyomdai pontrendszer, melynek kialakítását 1730 körül kezdte meg PIERRE-SIMON FOURNIER. Munkáját többen folytatták, köztük FIRMIN DIDOT [*didó*], aki 1770-ben a francia királyi láb 864-ed részeként határozta meg a nyomdászati pont méretét. Ezt aztán 1881-ben sikerült a méterrendszerhez igazítani: 1 méter = 2660 pont [10]. Így már könnyűszerrel kiszámíthatjuk 1 pont méretét, ami körülbelül 0,376 mm-nek adódik. Tekintettel a legszélesebb körben elterjedt szövegszerkesztő programok amerikai eredetére, meg kell említeni az angolszász mértékrendszert is. Itt a hüvelyk (inch) 72-ed részét sikerült pontméretnek választani. Ez a DIDOT-féle pont méretének 94% -a. Ma már inkább az angolszász rendszert használjuk, tekintettel a szövegszerkesztés és -szedés „elszámítógépesedésére”.

A betűk méretét csoportosíthatjuk felhasználásuk jellege szerint. Újságok, időszaki kiadványok cikkeinek főszövegéhez 8, 9 vagy 10 pontos betűfokozatot – ez valamelyest precízebb kifejezés, mint a betűméret – használnak, könyveket, esszéket pedig leggyakrabban 10, 11 vagy 12 pontos betűkből szednek. Dolgozatokhoz, levelekhez, beadványokhoz a 12-13 pontos betűfokozat tekinthető ideálisnak. Ennél is nagyobb méretben jelennek meg a kisgyermeknek szóló mesekönyvek. Érdekes, de egyben fontos is, hogy a különböző betűtípusok eltérő tényleges nagyságot mutatnak azonos pontméret mellett. „Azért, mert a betűméret a teljes sorszalagot jelenti a betűszárakkal együtt, amit azonban nagyságként érzékelünk, az a betűk középmagassága, s ez eltérő” [10, 63. oldal].

A választott betűfokozatnak a dokumentum jellegén kívül illeszkednie kell még a szöveg betűtípusához és annak adott változatához. Lényeges az is, hogy hosszú sorokhoz ne válasszunk túlon túl apró, rövid sorokhoz pedig nagyméretű betűt. Két- vagy többhasábos (azért sokkal több ne legyen...) tördelés esetén a 10-11 pont lehet a maximum.

## 2.4. Írásjelek

(Ehhez a szakaszhoz óravázlat és prezentáció található a szakdolgozat 1. számú mellékletében.)

### 2.4.1. Mondatvégi írásjelek

A magyar nyelvben a mondatok modalitásának kifejezésére *mondatvégi írásjeleket* – pontot, felkiáltójelet vagy kérdőjelet – használunk. Ezeket mindig közvetlenül az utolsó szó után, szóköz nélkül kell elhelyezni! Igazából csak a pontnak kellene szorosan „tapadni”, a másik két jelnek jót tenne egy egészen enyhe távolítás – még egyszer: nem szóköz! –, hogy ne tűnjön úgy, mintha az előtte álló szó egy karakterrel hosszabb lenne [10, 29. oldal]. Ha a szövegszerkesztőnk nem ennyire kifinomult, akkor érzük be nyugodtan a fent leírt elhelyezéssel.

A mondatvégi írásjel után viszont kell a szóköz. Külön említést érdemel a gondolat befejezetlenségét hangsúlyozó, illetve annak továbbgondolására ösztönző írásjel, a *három pont*. Ilyet használunk akkor is, ha egy részt kihagyunk a szövegből, illetve számok felsorolásából. Az elnevezés kicsit megtévesztő lehet: ez valójában egyetlen karakter! Három külön pontként begépelni teljesen rossz, mert túl sűrű lesz. Az sem megoldás, ha szóközöket helyezünk el közöttük, mert így fellazítjuk a szövegfolt egységét. Arról nem is beszélve, hogy a három darab pont könnyen áldozatául eshet egy sortörésnek, és egy vagy két pont átkerülése a következő sorba végképp indokolhatatlan volna. A helyzet abban az esetben sem reménytelen, ha számítógépünk billentyűzetén nem találjuk a megfelelő írásjelet: a legtöbb szerkesztőprogramban speciális karakterként vagy parancs segítségével könnyedén beszúrhatjuk a megfelelő helyre.

### 2.4.2. Központozás, idézetek

Mondaton belül a központozás jelzésére *vesszőt, pontosvesszőt vagy kettőspontot* alkalmazhatunk. Ezek elé soha nem gépelünk szóközt, utánuk viszont mindig: például így! Ugyancsak szorosan tapadnak a szöveghez a *zárójel*: nyitó zárójel után és záró zárójel előtt soha nincs szóköz. Ez a kerek,

a szögletes és a kapcsos zárójelekre egyaránt igaz. Iménti szabályunk sajátos helyzetet eredményez akkor, ha egy zárójeles megjegyzés végén szmájlit próbálunk elhelyezni: a záró zárójelet könnyű nevetésfokozó jelnek „olvasni”.

Az *idézőjelek* a magyar nyelvben „99” alakúak, tartalmukat a zárójelekhez hasonlóan szorosan zárják. [12, 26. oldal]. Alig túloznánk egyébként, ha azt mondanánk, hogy ahány nyelv, annyi alakú idézőjel. Még az angolon belül is másilyen alakú jellel idéznek a britek és az amerikaiak. Nem egyszerűbb az idézőjelek egymásba ágyazásának kérdése sem, de e helyütt elegendő csak a magyarban érvényes szabályra szorítkozni: „ha az imént említett idézőjelek között idézünk, akkor a »hegyével befelé forduló jelpár« jelenthet megoldást.” [4, 12. oldal].

### 2.4.3. Kis vízszintes vonalak

Kis vízszintes vonalakat több okból is el szoktunk helyezni dokumentumainkban: szükségünk lehet *kötőjelre*, *elválasztójelre*, *nagykötőjelre*, *gondolatjelre*, *mínusz jelre*, illetve a *kivonás jelére*. Igényes megjelenésű kiadványban nem fordulhat elő, hogy ezek mindegyikére ugyanazt az egyetlen karaktert használjuk. A kötőjel és a gondolatjel hossza különböző. Szavak elválasztásánál, ilyen-olyan szókapcsolatoknál, egy-két példa erejéig kötőjelet írunk. Ugyancsak kötőjellel kell rövidítésekhez ragot illeszteni, illetve személynévhez köthető fogalmat, mint például a Bernoulli-törvény, személynévhez kötni. Nagy kötőjel kell ellentétpárok, szerzőpárosok, intervallumok jelölésére (téli-nyári gumi, SZÖRÉNYI–BRÓDY, a sorozat 111–222. epizódjai).

A gondolatjel – melynek hossza, külalakja megegyezik a nagy kötőjellel – egyik alkalmazási területe a mellékmondatok közbeszúrása: mint az imént. Ilyenkor a gondolatjel elé és mögé is szóköz kell! Ugyancsak gondolatjelet használunk párbeszédnek, beszélgetések jelölésére. Ilyenkor minden szereplő minden megszólalását gondolatjel vezet be, általában új sorban. Ebben az esetben a gondolatjelet kicsi, de állandó szélességű szóköz követi. Ez azért fontos, mert csak így garantálható, hogy a jelet követő első betűk biztosan egymás alá kerüljenek.

A negatív számok előjele és a kivonás nevű matematikai művelet jele

megegyezik, csak míg előbbi szorosan tapad a jelölt számhoz, addig utóbbinál kell egy kicsi szóközt hagyni. A „mínuszjel” általában ugyanolyan hosszú, mint a gondolatjel, de alakja is és függőleges pozíciója is más: ez egy külön karakter.

A fent felsorolt jelek helytelen használata akár értelemzavaró is lehet. Egészen más a jelentése például annak, hogy „65-70. oldal” (valahol a hatvanötödik és a hetvenedik oldal között, vagy ezek környékén), vagy „65–70. oldal” (a hatvanötödik oldaltól a hetvenedik oldalig), illetve „65 – 70 oldalas dokumentum” (ez így mínusz öt oldalas szöveget jelentene, ha volna értelme – de nincs).

## 2.5. Üres helyek

Van olyan nézet, amely szerint egy dokumentumban a betűket (de legalábbis a szöveget) a szerző hozza, a tipográfus „csak” az üres helyeket adja hozzá. Könnyen belátható, hogy a szöveg olvashatóságát nem csekély mértékben befolyásolják a betűk, szavak, mondatok, sorok, bekezdések közötti távolságok, kihagyások. Vegyük sorra ezeknek az üres helyeknek a fajtáit!

### 2.5.1. Betűköz

A betűk egymás mellé helyezésekor ún. *betűközöket* hagyunk közöttük, hiszen ha nem így tennénk, akkor az olvasó szeme által már-már szétválogathatatlanul sűrű betűfolyamot kapnánk. Szerencsénkre a számítógépes szövegszerkesztő programokban fellelhető betűcsaládokban a betűközöket készen kapjuk, ami a hétköznapi iskolai használatra megfelelőnek mondható, és csak a professzionális kiadványkészítésben van szükség a betűközök finomhangolására, melyet egalizálásnak nevezünk. Mindazonáltal a betűközök egyenletessége alapvető esztétikai és olvashatósági követelmény [10]. Különösen a nagybetűknél eredményezne rossz ritmust, kellemetlen látványt, ha szigorúan és mechanikusan egyenlő távolságokra helyeznénk el azokat. Hasonló a helyzet nagybetű és kisbetű találkozásánál is. Ha a szerkesztendő dokumentumunkban kirívó egyenletlenségeket látunk – és temérdek időnk van –, akkor



2.1. ábra. Egalizálás

azért igazítsunk a betűközökön.

### 2.5.2. Szóköz

A *szóközök* kapcsán meg kell említeni egy újabb tipográfiai mértékegységet, ez pedig a *négyzet*, vagy idegen szóval kvirt. Relatív mértékegység, ami azt jelenti, hogy tényleges mérete függ attól, hogy milyen szöveggörnyezetben használjuk, ugyanis éppen az aktuálisan alkalmazott nagy M betű szélességével egyezik meg. Az „aktuális” alatt az adott méretű és adott változatú betűt értjük. A négyzet méretét jól közelítjük akkor, ha a betűfokozattal tekintjük egyenlőnek. Ezt valamelyest az elnevezés is sugallja, ha jobban belegondolunk.

Az idők során kialakult, és mára széles körben elfogadott gyakorlat szerint a szóközöknek 1/4 és 1/3 négyzet közé kell esniük. Hogy milyen méretű szóközöt tekintünk optimálisnak, az persze a történelem folyamán folyamatosan változott. GUTENBERG még – könnyen belátható, hogy ebben a témában az ő munkássága jelenti az időszámítás kezdetét – egészen szűk szóközöket szedett, de ezután egy növekedési folyamat indult meg, és tartott egészen az 1900-as évek végéig. A szóközök minden határon túli növekedése nem csupán esztétikailag eredményez elfogadhatatlan tipográfiát, hanem a papírtakarékos szedés szemléletének is ellentmond.

Manapság a számítógépes dokumentumszerkesztésben nagyon könnyű elérni, hogy a szöveg bal és jobb széle egyaránt a megfelelő margóhoz igazodjon. Nem kétséges, hogy az így igazított sorok, bekezdések, oldalak kellemes és rendezett látványt nyújtanak. A programok ezt a hatást a szóközök méretének dinamikus változtatásával érik el, ami viszont – különösen rövid sorok

esetén – erős túlzásokhoz vezethet. Ilyenkor jobban járunk, ha balra igazított sorokat használunk, amit szabadsoros szedésnek is neveznek. A szóközöknek egy speciális fajtája a *nem törhető szóköz*. Olyankor alkalmazzuk, amikor a szóköz két oldalán álló szavakat garantáltan egy sorban szeretnénk látni. A „8. parancsolat” vagy „IV. Béla” ilyen eseteket példáznak.

A szóközökről szólva végezetül még egy dolgot feltétlenül meg kell említeni. Vízszintes helykihagyást a szóközök *darabszámának* változtatásával érni el – magyarul egymás után több szóközt leütni –, nos, ez már a tipográfiai bűncselekmény kategóriájába tartozna (ha volna ilyen).

### 2.5.3. Sorköz

Az egymást túl sűrűn követő sorok nehezítik az olvasást, az erősen megritkított sorok pedig egy hosszabb dokumentum esetén komoly mértékű papírpazarláshoz vezethetnek. És szintén nem segítik az olvashatóságot, mivel a szemünk nehezebben találja meg az előzőtől távol lévő következő sort. Ismét a hagyományra, a kialakult gyakorlatra hivatkozunk, amikor azt mondjuk, hogy a betűfokozathoz képest kb. 20%-os ritkítást lehet optimálisnak tekinteni. Tizes betűmérethez például 12-es sortávolság illik, tehát két szomszédos sor megfelelő szélei közötti távolság 12 pont.

## 2.6. Szövegközi kiemelés

Nem ritkán fordul elő, hogy a folyószövegben bizonyos dolgokat, melyeket lényegesnek gondolunk, a szöveggörnyezettől el kell különíteni, ki kell emelni. Ez az eljárás a *szövegközi kiemelés*. Lényegében az adott betűtípus betűváltozatainak variálását jelenti. Az alkalmazás során egyidejűleg több szempontnak is meg kell felelni.

- Egyensúly. Túl sok kiemeléssel a szöveg összképe zavaros lesz, áttekinthetlenné válik, szétesik. Ha pedig túlságosan kevésszer nyúlunk ehhez az eszközhöz, akkor a céljának nem fog tudni megfelelni, hogy tudniillik könnyebben értelmezhetővé tegye a szöveget, mégpedig azáltal, hogy áttekinthetőbbé teszi [1].

- Célzatosság. Átgondolatlanul, céltalanul használni szövegek közötti kiemelést értelmetlen, ugyanúgy zsákutcába vezet, mint az előbb említett egyensúly felbillenése. A kiemelés célja alapvetően a hangsúlyozás, a rendszerezés vagy a megkülönböztetés lehet [1].
- Következetesség. Fontos, hogy a kiemelések rendszere, stílusa egységes maradjon végig a dokumentum keretein belül. Célszerű a szöveg formázásának megkezdése előtt végiggondolni, hogy mit és milyen módon szeretnénk kiemelni.

### 2.6.1. Nem javasolt módozatok szövegek közötti kiemelésre

Dokumentumunk folyószövegében elszórtan használni kiemelés céljából a félkövér betűváltozatot nem szerencsés, mert feltűnő sötét foltokat képez, megbontva ezzel az adott oldal harmonikus kinézetét. (Más esetben azonban lehet létjogosultsága ennek a módozatnak is – erről hamarosan.) Nem szabad alkalmazni betűritkítást sem, de nem is érdemes: kiemelő értéke csekély.

Teljesen rossz gondolat csupa nagybetűs – szakkifejezéssel élve: verzál – kiemeléssel élni. A sortávolság szempontjából vezetne elfogadhatatlan eredményre. Felettük túl szűk, alattuk túl bő lenne a hely. Ezt mondják úgy is, hogy „fönnragadnak” [10]. Abban megoszlanak a vélemények, hogy az imént említett módozatok használhatók-e a folyószöveg mondatain belül. Van, aki semmi kivetnivalót nem lát például a ritkított betűs vagy a vastagítással történő kiemelésben [8], míg mások – [1], [10], [4] – szerint ez határozottan ellenjavallt. Amint az fentebb kiderült, jómagam utóbbiak táborához tartozom, abszolút zsinórmértéknek pedig VIRÁGVÖLGYI PÉTER művét [10] tekintem.

Létezik azonban egy olyan módozat szövegek közötti kiemelésre, amelyben valamennyi vonatkozó forrásom teljesen egységes álláspontot képvisel, miszerint nagyon rossz: ez pedig az aláhúzás. Nyomatott szöveg olvasásakor az információt jobbra a betűk felső részéből nyerjük, mivel az egyedibb. Szemünk, agyunk ehhez szokott hozzá, ehhez alkalmazkodik. Az aláhúzás viszont a betűk alsó széléhez vonzza a tekintetünket, amivel alaposan megnehezíti az

olvasott szöveg feldolgozását, megértését. Sehol, semmilyen aláhúzást ne alkalmazzunk!

### 2.6.2. Javasolt módozatok szövegközi kiemelésre

Szövegközi kiemelésre kétféle betűváltozatot helyes használni: a kurzívet és a kiskapitálist. A *kurzív* – más néven italic vagy dőlt – betűnek nem csupán a dőlésszöge különbözik az állótól, de a rajzolata is más. Ezáltal jól kiemelkedik a szövegből, ugyanakkor nem rontja el az összképet. Folyószövegben a létező legjobb kiemelési módozat, de alkalmazható képaláírások, címek szedésére is. Élőfejben ugyancsak jól mutat.

A kiskapitális betűváltozat alapvetően személynevek kiemelésére használható, elsősorban nem szépirodalmi jellegű munkákban. Színdarabok esetén a szereplők, irodalomjegyzékekben pedig a szerzők neveit célszerű ilyen módon kiemelni. Ez a változat mindenhol jól illeszkedik a szövegbe – különösen, ha valódi kiskapitálist használunk.

Végül definíciók, tételek, címszavak kiemelésére jó megoldás a vastag betűs, vagy más néven félkövér szedés. Ehhez az kell, hogy ezek mindig a sor elején és egymás alatt álljanak. Ilyenkor ugyanis hatékonyan fókuszálják a szemet a kiemelt szóra, de nem tesznek rosszat a szöveg látványával.

## 2.7. Tördelés

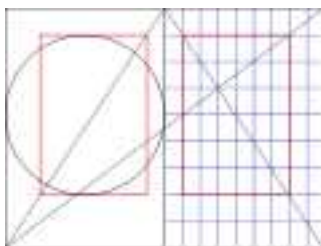
Tördelésen alapvetően két dolgot értünk. Egyrészt a hosszabb szöveg oldalnyi adagokba rendezését, másrészt bármely típusú dokumentumban az alkotórészek végleges összerendezését. Ilyen alkotórészek: a főszöveg, a jegyzetek (láb-, vég-, széljegyzetek), illusztrációk (képek, ábrák, táblázatok). A tördelés történhet egy vagy több hasábra, sőt, ezeket kombinálni is szabad.

### 2.7.1. Margók, szedéstükör

A *szedéstükör* a lapnak az a része, ahova a nyomtatott főszöveg kerülhet. Mellette kétoldalt, valamint alatta és fölötte helyezkednek el a *margók*. Egy-

oldalas szedésnél a jobb oldali, kétoldalasnál a külső margóra kerülhet a széljegyzet, ha dokumentumunk tartalmaz ilyet. (A széljegyzet műfajának örök klasszikusa PIERRE FERMAT XVII. századi francia matematikus, aki a – saját maga által generált – legenda szerint csak azért nem írta le híres sejtésének bizonyítását, mert az nem fért volna el az éppen olvasott könyv margóján. Sikerült ezzel évszázadokig sokkolnia a matematikus társadalmat, hiszen nagyon sokáig senki nem tudott bizonyítást találni erre az amúgy nagyon egyszerűen megfogalmazható problémára. Néhány éve a helyzet rendeződött, és azóta a nagy Fermat-sejtést Wiles-tételnek nevezik.)

A *margók* méretére vonatkozó szakmai ajánlás nagyjából annyi, hogy izléses legyen. Modern szedésű kétoldalás dokumentumokhoz egy javaslat: legyen 2 cm-es a belső, 4 cm-es a külső, 3 cm-es a felső és 5 vagy 6 cm-es az alsó margó. Érdekességképp egy középkori szedéstükör-változat látható az 2.2. ábrán. Megvalósítani például a téglalap oldalainak kilenc részre



2.2. ábra. Szedéstükör

osztásával lehetett.

Bármilyen szedéstükröt választunk, arra mindenképpen ügyelni kell, hogy a külső margó – ez egyoldalás szedésnél mindig a jobb oldali – elég széles legyen ahhoz, hogy a nyomtatott dokumentumot kézbe lehessen fogni anélkül, hogy ez az olvashatóságot zavarná, azaz maradjon elég széles hely a lapot tartó hüvelykujjainak. Ha a mű széljegyzetet is tartalmaz, akkor természetesen a margók méretének megválasztásakor ezt figyelembe kell venni.

### 2.7.2. Címrendszer

GYURGYÁK JÁNOS így ír erről a Szerkesztők és szerzők kézikönyvében [3]: „A tagolás elsődleges eszközei a belső címek. A címek a folyószöveget szö-

vegrészekre, szövegegységekre bontják, segítségükkel feltárul a könyv szerkezete, belső logikája – vagy ellenkezőleg, komplikált, zavaros felépítése. Csak nagyon kivételes esetben adjunk hosszú, magyarázó belső címeket. A magyarázat, a kiegészítés nem a cím, hanem az azt követő szöveg feladata, ne akarjunk minden kérdést már a címbe megoldani. A terjedelmes címek elkedvetlenítik az olvasót, nem is beszélve a tipográfiai problémákról. A műben az azonos rangú részegységeket azonos fokozatú címekkel látjuk el.

A részegységek további tagolására alacsonyabb rendű címeket iktatunk a szövegbe, a kisebb részeket ismét tovább tagolhatjuk stb. Így jelenik meg, mintegy a részegységek vázlataként a címrendszer.” Elvi lehetőségként felmerülhetne, hogy a címeket a folyószövegétől eltérő betűtípusból szedjük, de mivel a betűkeverés nehéz és problémás feladat, ezért válasszunk egyszerűbb és legalább olyan jó megoldást: a betűváltozatokat.

Szabályok.

- Azonos szintű címeket minden esetben azonos betűfajtaival írjunk és a betűfokozatuk is legyen állandó.
- Cím után soha ne tegyünk pontot. Akkor se, ha ún. bekezdéscímről van szó, ami azt jelenti, hogy a folyószöveg vele egy sorban folytatódik. Különítsük el például félkövér vagy kiskapitális betűkkel. Esetleg használhatjuk a középmagas pont nevű jelet.
- Alacsonyabb rendű cím fölött mindig több térköznek kell lenni, mint alatta: érzékeltessük, hogy az utána következő szöveghez tartozik [10].
- A cím mindig legyen rövid, tömör. A magyarázatot az olvasónak a címet követő szövegrészben kell megtalálnia, nem pedig magában a címbe.

A részek, fejezetek, alfejezetek rendszerét teljesen felesleges 4-5 fokozatnál jobban elmélyíteni. A cél az áttekinthetőség mértékének növelése, a világosabbá tétel, nem pedig a túlbonyolítás. Szakmai jellegű műveknél élhetünk a címek sorszámozásának lehetőségével is, bár ezzel nem minden forrás ért egyet: [3] szerint ezek csupán mankók, amelyek – jó tipográfia mellett – nem

teszik világosabbá a dokumentum szerkezetét, tehát feleslegesek. Ha azonban a szöveg fejezetekre, szakaszokra stb. vonatkozó utalásokat tartalmaz, akkor véleményem szerint jó szolgálatot tehetnek a címekhez rendelt sorszámok.

A címrendszer és a címrendszer hierarchiája a tartalomjegyzékben is megjelenik, pontosan követve a főszövegben kialakított rendszert, de tipográfiai-lag nem szolgálai módon másolva azt.

A címrendszerre vonatkozó fontos tördelési szempont, hogy oldal vagy hasáb aljára nem kerülhet cím. Legalább három sornak kell még következnie. Ha ez nem oldható meg, akkor inkább vigyük át a címet is a következő oldalra vagy hasábra.

## 2.8. Olvashatóság

Valójában valamennyi eddig ismertetett tipográfiai ajánlás, szempont, szabály legfőbb célja a dokumentum olvashatóságának, és ezáltal megértésének elősegítése. Kísérletek igazolják, hogy a két dolog között szoros összefüggés van. Hosszabb szövegek folyamatos olvasására csak a kisbetűs szedés alkalmas, mivel ebben a formában a szavaknak jellegzetesebb sziluettjük van az ékezetek és a felnyúló-lenyúló szárak miatt [10].

A csupa nagybetűs szöveg 10-12%-kal lassabban olvasható. Különösen webes publikálás esetén nem elhanyagolható szempont az sem, hogy az instant messenger-programok korában a verzál (csupa nagybetűs) szedés az üvöltözés egyezményes jele.

Nyomtatott dokumentumokban célszerű a változó vonalvastagságú, talpas betűtípusok közül választani, elektronikus megjelenéshez viszont jobb egy sans serif és az átlagosnál nagyobb betűszemű betű (ld. a 2.3. szakaszt a 13. oldalon). „Az olvasás az egyik legkonzervatívabb tevékenység a világon. Ismerős alakzatokat várunk el, s nem toleráljuk a »meglepetéseket«. Ezt úgy is meg lehet fogalmazni, hogy a betűtípus addig jó, amíg észrevétlen – ám ha a különleges formák előtérbe tolnakodnak... , nem tudunk a szöveg értelmére figyelni.” [10]

Fontos az is, hogy milyen hosszú sorokba tördeljük a folyószöveget. Egyszerű, egyhasábos dokumentum esetén ügyeljünk arra, hogy 40-nél kevesebb

és 80-nál több karakter ne kerüljön egy sorba. Mindezek mellett hosszú szövegnél még figyelni kell(ene) arra is, hogy nyomtatáshoz a lehető legfeketébb festéket használjuk, hogy a papír, amelyre nyomtatunk fehér vagy krémszínű, de mindeképpen matt felületű legyen, és hogy megfelelően erős megvilágítás mellett olvassunk.

## 3. fejezet

# Szövegszerkesztés $\text{\LaTeX}$ környezetben

### 3.1. Alapok

Az első részben tárgyalt tipográfiai hagyományoknak, rendszabályoknak minden tekintetben megfelelni nem könnyű. A középiskolai tananyaghoz kapcsolódó valamelyik „hagyományos” WYSIWYG<sup>1</sup> szövegszerkesztővel hosszabb dokumentumot elkészíteni úgy, hogy az tipográfiai (is) igényes legyen, kifejezetten macerás. (Hogy erre miért van egyáltalán szükség, arról többek között a 2.8. szakaszban olvashatunk a 26. oldalon.)

Üdvös volna, ha létezne egy olyan dokumentumformázó rendszer, amely alapból, gyárilag „tudja” a tipográfiát, de amely természetesen engedné, hogy tetszőleges mértékben eltérjünk a szigorú szövegszedési szabályoktól, ha úgy tartja fényes úri kedvünk. És amellyel minden tekintetben nyomdai minőségű kiadványokat állíthatnánk elő. Esetleg nem csupán egyszerű szövegek szerkesztésére, meg képek, ábrák, táblázatok beillesztésére volna alkalmas, de professzionális színvonalon támogatná matematikai formulák, kémiai szerkezeti képletek létrehozását is. Aztán, ha már ennyi mindent tudna, akkor amolyan extra felszereltségként könnyedén megbirkózhatna olyan különleges feladatokkal is, mint például a kották vagy a sakkállások szedése.

---

<sup>1</sup>What You See Is What You Get – Amit látsz, azt kapod!

Nos, a jó hír az, hogy bő harminc éve létezik ilyen rendszer. Az amerikai Stanford egyetemen alkotta meg DONALD E. KNUTH matematikus. A legenda szerint azért, mert finoman szólva elégedetlen volt „A számítógép-programozás művészete” című könyvének nyomdai előnézetével, vagy szakszerűbben: kefelevonatával. Nekiült hát, és készített egy saját szövegszedő rendszert. Egy olyat, amely alkalmas arra, hogy a fentebb leírt feltételes mondatokat egy csapásra kijelentő módba tegye [6].

Ez lett a  $\TeX$ .

### 3.1.1. A $\TeX$ és a $\LaTeX$

A  $\TeX$  tehát egy szövegszedő program, amellyel a legmagasabb tipográfiai igényeket is kielégítő módon, nyomdai minőségben készíthető el gyakorlatilag tetszőleges típusú írásos anyag: a levéltől kezdve a tudományos cikkeken és dolgozatokon át a könyvekig bármi. (Nem feledkezve meg a kottákról és a sakkállásokról sem...) A program a görög „művészet” –  $\tau\epsilon\chi\nu\eta$  – szó első három betűjéből keletkezett [11]. Ennélfogva ejteni úgy kell, mint a „számítástechnika” szót „számítás-” és „-nika” nélkül.

A  $\LaTeX$  pedig egy olyan programgyűjtemény, amely a  $\TeX$ -re épülve teszi lehetővé, hogy viszonylag kényelmesen – ez a kényelem mondjuk tényleg relatív, erről mindjárt a következő pontban – hozzassunk létre minden igényt kielégítő színvonalú dokumentumokat. Megalkotásába az 1980-as évek elején kezdett bele LESLIE LAMPORT. A programcsomag fejlesztésébe azóta sokan bekapcsolódtak.

Ennek a fejezetnek az a célja, hogy segítséget adjon a  $\LaTeX$  használatának alapfokú elsajátításához.

### 3.1.2. A $\LaTeX$ előnyei és hátrányai

Lássuk elsőként a hátrányokat!

- Leírás nélkül boldogulni a  $\LaTeX$ -hel nagyjából reménytelen. (Szerencsére ma már léteznek leírások, ha mégoly tömörek és rövidek is időnként, mint ez a fejezet.)

- Rövidebb – legfeljebb néhány oldalas – dokumentumok készítése gyorsabban oldható meg MS Word-del, OpenOffice Writer-rel, vagy bármely egyéb WYSIWYG rendszer segítségével.
- Ha meghatároztuk a kiadványunk szerkezetét, akkor a tipográfiai feladatokat a L<sup>A</sup>T<sub>E</sub>X hibátlanul és magas színvonalon oldja meg (ez mondjuk még nem hátrány...), de lényegesen, alapjaiban eltérni az előre definiált kinézetektől kimondottan nehéz.
- Az iskolában azt szoktuk meg, hogy gépelés közben nagyjából már látjuk is a munkánk eredményét (részünkről, tanárok részéről általában pusztába kiáltott szó marad, hogy formázni kizárólag a *begépelés után* kezdjük a dokumentumot – tisztelet a kevés számú kivételnek). A L<sup>A</sup>T<sub>E</sub>X-ben efféle kényelemről szó sincs.

De akkor miért érdemes mégis megtanulni a L<sup>A</sup>T<sub>E</sub>X használatát?

- A L<sup>A</sup>T<sub>E</sub>X-hel készült kiadványok tényleg úgy néznek ki, mintha nyomdában készültek volna!
- Nem kell aggódnunk azon, hogy tipográfiailag rendben van-e az a dokumentum, amit kiadunk a kezünkből.
- Néhány parancsot kell megjegyezni csupán, hogy alapszinten boldoguljunk, ugyanakkor magas szinten szövegszerkesszünk.
- A L<sup>A</sup>T<sub>E</sub>X-ben készült munkák abszolút hordozhatóak, maga a T<sub>E</sub>X és a L<sup>A</sup>T<sub>E</sub>X pedig szabadon (tehát ingyen) hozzáférhető a főbb operációs rendszerek mindegyikén.
- Matematikai képleteket, egyenleteket, formulákat könnyedén és hihetetlen minőségben készíthetünk vele.
- A bonyolultabb dokumentumrészek – hivatkozások, láb-, vég- és széljegyzetek, tartalomjegyzék, tárgymutató, irodalomjegyzék – könnyedén előállíthatóak, ugyanakkor hibátlan megjelenésűek és pontosak lesznek.

- A L<sup>A</sup>T<sub>E</sub>X logikus, rendezett gondolkodásra ösztönöz, mert a dokumentum szerkezetét pontosan le kell írni a program számára, ehhez pedig a struktúrát végig kell gondolni.
- Az érettségi után az egyetemen, főiskolán szinte biztosan – de legalábbis igen nagy valószínűséggel – szükség lesz a L<sup>A</sup>T<sub>E</sub>X használatára (különösen műszaki vagy természettudományos területen); akkor pedig komoly helyzeti előny lesz majd, ha nem a nulláról kell indulni.

### 3.1.3. Egy T<sub>E</sub>X rendszer telepítése

Ha mérlegeltük a hátrányokat és előnyöket, és úgy döntöttünk, hogy belevágunk a L<sup>A</sup>T<sub>E</sub>X megismerésébe, akkor először is szükségünk lesz egy T<sub>E</sub>X rendszerre. Ha otthon is az iskolában megszokott Windows-os környezetben dolgozunk, érdemes a MiK<sub>T</sub>E<sub>X</sub> nevű disztribúciót beszerezni és feltelepíteni. Ez a szakasz ehhez nyújt segítséget.

1. Látogassunk el a <http://miktex.org> weboldalra és válasszuk bal oldalt a „Download” kategóriában a MiK<sub>T</sub>E<sub>X</sub> 2.7 linket.
2. A lap alsó harmadában találjuk a MiK<sub>T</sub>E<sub>X</sub> 2.7 Net Installer-t. Töltsük le! (Néhány megabájt csupán.)
3. Indítsuk el a letöltött telepítőt!
4. Az első fontos kérdés, amire válaszolnunk kell, hogy kezdésként csak letölteni szeretnénk-e a Mik<sub>T</sub>E<sub>X</sub>-et (Download MikTeX rádiógomb), vagy rögtön installálni (Install MikTeX) – ez utóbbit semmiképpen ne válasszuk a letöltés előtt!
5. A következő eldöntendő, hogy Basic MikTeX vagy Complete MikTeX rendszert szeretnénk. Elegendő a Basic változat – a telepítőprogram is ezt javasolja –, ha van az iskolaihoz hasonló folyamatos, szélessávú internetelérésünk.
6. A letöltés végeztével indítsuk el újra a MikTeX telepítőjét. Ekkor már választhatjuk az „Install MikTeX” gombot.

7. Ismét választhatunk a „Basic” és a „Complete” között, de a telepítő emlékszik az előbbi választásunkra, és azt ajánlja fel.
8. Ha rendszergazdai jogosítványokkal felvértezve telepítünk, akkor meghatározhatjuk, hogy a többi felhasználó is használhatja-e a  $\TeX$ -ünket. („All Users”)
9. Válasszuk ki azt a könyvtárat, *ahonnan* telepíthet a gép – ez ugyanaz, *ahova* az előbb letöltöttük a programot és a csomagokat.
10. Most azt határozzuk meg, hogy *hol* lakjon gépünkön a  $\TeX$ .
11. Végül nyomjunk egy jól megérdemelt „Finish” gombot!

Használatba vétel előtt keressük meg a Start menüben a Programok MiKTeX csoportjában a „MiKTeX Options”-t. Az ekkor elének kerülő párbeszédpanel lapjai közül vegyük sorra azokat, amelyek számunkra fontosak lehetnek. A „General” lapon a „Package installation” részben válasszuk az „Ask me first” lehetőséget. Így majd egy hiányzó csomag letöltése előtt a rendszer kérdést szegez nekünk: akarjuk-e ezt valóban? A „Packages” lapon a „Language Support”-ban jelöljük be a „Hungarian”-t.

Ugyanezen a lapon a „Start Package Manager” gombra kattintva egy új ablakban láthatjuk az összes csomagot alfabetikus sorrendben. Kiderül az is, melyek nincsenek telepítve, illetve nagyon röviden az is, hogy melyik csomag mire szolgál.

Még mindig a „Packages” lapon, a „Package repository” rovatban be kell állítanunk, hogy honnan fogjuk telepíteni a hiányzó csomagokat. Amennyiben „Basic”-et telepítettünk, úgy ez egy webhely beállítását jelenti, míg ha a „Complete” mellett döntöttünk, akkor a csomagok már a gépünkön várnak a bevetésre, csak a helyüket kell beállítanunk, a „Local folder” gombjára, majd a „Tovább”-ra kattintva.

A mostanra már létező MiKTeX-ünk komfortos használatához feltétlenül töltsük le és telepítsük a TeXnicCenter nevű  $\LaTeX$ -burok programot. A  $\LaTeX$ -burok olyan szoftver, amely nem csupán szövegszerkesztésre alkalmas, hanem képes elvégezni a fordítást, a dokumentumaink megjelenítését, segédprogramok futtatását. A TeXnicCenter telepítője letölthető pl.

a <http://www.toolscenter.org/> oldalról, a „Download” gombra kattintva. Itt a „TeXnicCenter Installer”-t kell választani. A `TXCSetup_1Beta7_50.exe` letöltésével és telepítésével egy remek keretprogramot nyerünk L<sup>A</sup>T<sub>E</sub>X-ben születő dokumentumaink készítéséhez.

### 3.1.4. Egy L<sup>A</sup>T<sub>E</sub>X forrásállomány felépítése

Van már L<sup>A</sup>T<sub>E</sub>X-ünk és van hozzá a nyers szöveg és a kiadvány formátumát leíró parancsok bevitelére szolgáló szövegszerkesztőnk; kezdődhet a dokumentumok létrehozása. Először is kell hozzá egy forrásállomány. Ez lényegében a fent említett formátumleíró parancsokat és a mű nyers szövegét tartalmazza. Egy `.tex` kiterjesztésű fájl, amit a L<sup>A</sup>T<sub>E</sub>X le tud fordítani, és belőle az előre meghatározott fájlformátumban kimenetet készíteni. Ez a kimenet lesz a kész dokumentum.

Minden forrásállományban meg kell jelennie az alábbi három sornak:

```
\documentclass{article}
\begin{document}
  ide jön a főszöveg és a formátumleíró parancsok
\end{document}
```

Ezzel rögtön a forrás két fő részre tagolását is megoldottuk: a

```
\begin{document}
```

előtti rész a *preambulum*, ami utána következik, az pedig a *dokumentumtörzs*. A preambulumba olyan parancsok kerülnek, amelyek az egész dokumentumra vonatkoznak.

Ahhoz, hogy a készülő mű a magyar nyelv és a magyar tipográfia szabályainak minél jobban megfeleljen, szükség van még pár dologra a preambulumban, úgyhogy a forrásállományunk eleje inkább így nézzen ki:

```
\documentclass[a4paper, 12pt]{article}
\def\magyarOptions{defaults=hu-min}
\usepackage[magyar]{babel}
\usepackage{t1enc}
\usepackage[latin2]{inputenc}
```

```
\usepackage{fancyhdr}
\usepackage{setspace}

\begin{document}
  ide jön a főszöveg és a további formátumleíró parancsok
\end{document}
```

Ez most így nyilván egy rakás talány, viszont terjedelmi korlátok miatt nem nagyon merülhetünk el a részletekben. Később a fenti sorok azért valamelyest értelmet nyernek majd.

### 3.1.5. Begépelés, fordítás, kimeneti fájl

Szövegeink begépelésekor néhány – eleinte szokatlannak tűnő – szabályt be kell tartanunk.

**A sor elején lévő szóközök** nem számítanak. Szavak, írásjelek közé akárhányszóközöt írunk is, az egynek fog számítani. (Így aztán elvileg sem lehet elkövetni a „vízszintes helykihagyás szóközökkel” nevű tipográfiai bűncselekményt.) Ha olyan szóközöt szeretnénk, ami garantáltan nem fog sortörés áldozatául esni, akkor a ~ jelet kell begépelnünk az adott helyre. Például: IV.~Béla, 8.~parancsolat vagy 10~cm. Szóköznek számít a sorvége karakter – „Enter” – is, ha csak egy van belőle.

**A L<sup>A</sup>T<sub>E</sub>X rendkívül igényesen tördel;** a legjobb, ha ráhagyjuk, hogy hol kezdődjön új sor, mi pedig csak gépelünk folyamatosan. Ha mégis szándékosan új sort akarunk kezdeni, akkor ezt a \newline paranccsal, vagy két \\ jel begépelésével érhetjük el. Ilyenkor új bekezdés nem kezdődik. (Új bekezdést kettő darab „Enter”-rel kezdhetünk – ez a forrásállomány szövegében egy üres sornak felel meg.) Oldaltörést a \newpage paranccsal hozhatunk létre.

**Az igényes tördeléshez** szükség van a szavak helyes elválasztására. A L<sup>A</sup>T<sub>E</sub>X-nek ez sem gond, de ha mégis hibát vét, vagy kivételes eseteket akarunk megtanítani neki, akkor a forrásállomány preambulumban kiadhatjuk a \hyphenation{sza-vak lis-tá-ja} parancsot, ahol

a szólistában szereplő szavak lehetséges elválasztási helyeit jelölhetjük „-” jelekkel.

**Van néhány speciális karakter**, amit a L<sup>A</sup>T<sub>E</sub>X lefoglalt magának, így ezeket a szövegben csak némi ügyeskedéssel használhatjuk. Ilyen jelek: `\ { } ~ % $ & # ^ _ A \` például a parancsok kezdőjele, a `$`-ral válthatunk matematikai módba, a `%`-kal pedig azt mondjuk a rendszernek, hogy az utána következő szöveget egészen a sor végéig hagyja figyelmen kívül a forrásállomány lefordításakor, mert mondjuk az csak egy szerkesztői megjegyzés. Alakulhat úgy, hogy ezek valamelyikére mégis szükség van a szövegben – elég csak a `%` jelre gondolni. Ilyenkor a foglalt karakter elé egy `\` jelet teszünk, és máris megjelenik a dokumentumunkban. (Magát a `\` jelet pedig így írhatjuk ki: `$\backslash$`.)

**Kötőjelet** a billentyűzet „-” jelével gépelünk be, nagykötőjelet és gondolatjelet két ilyen jel egymás utánjával: „--”, mínusz jelet pedig így: `$-$`. Ezen jeleket helyesen kell használni, amint arról a 2.4.3. szakaszban olvashattunk a 18. oldalon.

**A három pont** egyetlen – igaz, speciális – karakter. (2.4.1. szakasz, 17. oldal.) Megjelenítése a `\dots` paranccsal történik.

**Az idézőjeleket** nem szabad a billentyűzeten megtalálható "-jellel begépelni! A nyitó idézőjelet két vesszővel, a zárót két aposztróffal (általában Shift+1) vihetjük be. Jó megoldás még a `\textqq{idézőjelbe kerülő szöveg}` parancs is.

Dokumentumunk lényegi részének, a folyószövegnek a begépelésekor elemi és jogos igényünk, hogy a magyar ábécé összes betűjét gond nélkül, a szokásos 101 gombos billentyűzetről vihessük be. Erre jelent megoldást a preambulumban a `\usepackage[latin2]{inputenc}` sor. Hogy minden rendben van-e az ékezetes karakterek terén, azt az „árvíztűrő tükörfúrógép” – mérsékeltén értelmes – kifejezéssel vagy az „Öt szép szűzlány örült írót nyúz.” mondattal tesztelhetjük. Ugyancsak a L<sup>A</sup>T<sub>E</sub>X magyarítása érdekében kell beírni a

```
\def\magyarOptions{defaults=hu-min}
```

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

sorokat. Fentiekről részletesen olvashatunk [11]-ben. (Közbevetőleg egy apró tipp a  $\LaTeX$ -hel való ismerkedéshez: tetszőleges mennyiségű „folyószöveget” állíthatunk elő bármelyik, weben fellelhető Lorem ipsum-generátorral; használhatjuk például a <http://hu.lipsum.com/> oldalt. Hogy mi is az a Lorem ipsum, annak pedig a Wikipédián utána lehet nézni.)

Van tehát már forrásállományunk, de nincs még igényes tipográfiájú, nyomtatható dokumentumunk. Ahhoz, hogy ilyen keletkezzen, a forrást – ami egy `.tex` kiterjesztésű fájl – le kell fordítani. A TeXnicCenter-ben látszik egy legördíthető lista, alapesetben a `LaTeX => DVI` beállítással. Ez arra utal, hogy a  $\LaTeX$  a fordítás során egy eszközfüggetlen (DeVice Independent) nyomtatható fájlt fog előállítani, amit a TeXnicCenter-rel együtt települő Yap nevű programmal lehet nézegetni, és ha mindent rendben találtunk – vagy lejárt a határidő... –, nyomtatóra küldeni. Arra is van lehetőség, hogy a kimenet ne `.dvi`, hanem `.pdf` fájl legyen. Erről a 3.6 szakaszban még lesz szó.

Fordítani és kimeneti fájlt készíteni a `Ctrl+F7` billentyűkombinációval lehet. Ha a lefordított fájlt rögtön meg is szeretnénk tekinteni, akkor nyomjuk le egyszerre a `Ctrl+Shift+F5` gombokat. Mindkét lehetőség elérhető természetesen menüből is. (Build  $\rightarrow$  Current File)

## 3.2. A kiadvány tipográfiájának beállítása

Ahhoz, hogy egy dokumentumot nyomdai minőségben tudjunk előállítani, pontosan meg kell határoznunk néhány fontos dolgot. Milyen típusú kiadványt szerkesztünk éppen, milyen legyen az oldalak általános kinézete, milyen betűtípust választunk, stb. Az egész művet érintő alapvető beállítások helye általában a preambulumban van.

### 3.2.1. Dokumentumosztályok

Az első, amire a  $\LaTeX$  kíváncsi, az a szerkesztendő dokumentum típusa. Ezt a `\documentclass{osztály}` paranccsal állíthatjuk be. Az „osztály” szó he-

lyére a következőket írhatjuk:

**article** Ezt választjuk, ha cikket, jelentést, feljegyzést, témazáró dolgozatot, bármilyen kisebb – néhány oldalas – dokumentumot készítünk. A leggyakrabban használt osztály.

**report** Hosszabb tanulmányok, OKTV<sup>2</sup> pályamunkák, szakdolgozatok szerkesztéséhez.

**book** Ha könyvet írunk, ez a mi osztályunk.

**letter** Szabványos és szabályos – főleg hivatalos – leveleket írhatunk ezzel a dokumentumosztállyal.

**slides** A PowerPoint alternatívája: prezentációkhoz készíthetünk vele fóliákat.

A parancshoz opciókat is rendelhetünk, mintegy finomhangolva a választott dokumentumosztály működését. Ekkor az általános alak

`\documentclass[opciók]{osztály}`-ra változik. Az opciók segítségével állíthatjuk be az alapértelmezettől eltérő papírméretet, betűfokozatot, hasábszámot és még egy sereg egyéb dolgot. Ezen dolgok felsorolását megtaláljuk a [11] 185. oldalán, vagy a [7] 8. oldalán. Itt csak két példát említünk meg: `\documentclass[a4paper, 12pt]{report}` – A/4-es papíron, 12-es betűmérettel például szakdolgozatot írunk.

`\documentclass[a4paper, 10pt, twocolumn]{article}` – az iskolaújság számára alkotunk egy kéthasábos cikket, 10 pontos betűkkel.

### 3.2.2. Oldalformátum

Lehetőségünk van a kiadványunkat kétoldalásra tervezni. Ehhez *article* osztály esetén fel kell vennünk a `\documentclass` parancs opciói közé a `twoside`

---

<sup>2</sup>Országos Középiskolai Tanulmányi Verseny

paramétert. A kétoldalas szedés azt jelenti, hogy a kinyomtatott szöveg megjelenése függ attól, hogy páros vagy páratlan oldalra került-e. (A *book* osztályú könyvek alapértelmezés szerint így készülnek.) Ilyenkor a lap jobb és bal oldala helyett helyesebb a külső és belső széle kifejezéseket használni.

Egy átlagos oldal a főszövegben három részből áll: fejléc, lábléc, folyószöveg. Az oldalszám helye például a láblécben vagy a fejlécben van. Ugyancsak a fejlécbe kerül az ún. élőfej, ami tetszőleges tartalmú lehet, de legtöbbször cím, felirat kerül bele tájékoztató jelleggel, a dokumentumban való eligazodást megkönnyítendő. Az sem ritka, ha úgy a fejléc, mint a lábléc üresen marad.

A L<sup>A</sup>T<sub>E</sub>X-ben az oldal stílusát meghatározni annyit jelent csupán, hogy megmondjuk, mi kerüljön a fejlécbe és a láblécbe [11]. Meghatározhatjuk általánosan, az egész dokumentumra vonatkozóan, vagy csak egy adott oldal erejéig. Előbbihez a `\pagestyle`, utóbbihoz a `\thispagestyle` parancs használható. A `\pagestyle{empty}` hatására minden oldalon üresen fog maradni úgy a lábléc, mint a fejléc. `\pagestyle{plain}` használatával azt érjük el, hogy a láblécbe bekerül az oldalszám, a fejlécbe pedig semmi. A `\pagestyle{headings}`-szel viszont a lábléc marad üres, míg a fejlécbe a mű címe vagy a fejezet címe – dokumentumosztálytól függően. Egyéni élőfej készítésére is van lehetőség, a `\pagestyle{myheadings}` segítségével.

### 3.2.3. Betűk

A L<sup>A</sup>T<sub>E</sub>X alapértelmezés szerinti betűcsaládja a Computer Modern. Jól olvasható, szép betűtípus, nincs okunk leváltani. Matematikai tárgyú szövegeknél különösen ragaszkodnunk kell hozzá. Betűváltozatok terén a L<sup>A</sup>T<sub>E</sub>X – a tőle elvárható módon – jól áll. Álló, döntött, dőlt, félkövér, kiskapitális és írógép típusú betűkkel egyaránt gazdálkodhatunk. Használatukhoz a TeXnicCenter-ben ikonok és menüpontok (Format → Characters) egyaránt rendelkezésre állnak. Ha nagyon ráérünk, természetesen ezeket a betűparancsokat kézzel is beírhatjuk a forrásállományba.

```
\textsl{szöveg} döntött betűk
```

```
\textit{szöveg} dőlt betűk
```

```
\textbf{szöveg} félkövér betűk
\textsc{szöveg} KISKAPITÁLIS BETŰK
\texttt{szöveg} írógép stílusú betűk
```

A döntött betűk a magyar tipográfiában nem elfogadottak.

Szövegek közötti kiemelésre használjuk az `\emph{szöveg}` parancsot. Alap esetben ez is dőlt – tehát nem döntött – betűket eredményez, csak annyival praktikusabb, hogy több ilyen egymásba ágyazva a kiemelt szövegen belüli kiemeléseket is helyesen kezeli. Például:

*így néz ki, amikor kiemelt szövegen belül alkalmazunk kiemelést.*

### 3.3. A szöveg szedése

#### 3.3.1. Környezetek

„Egy megadott funkció elvégzésére szolgáló, névvel ellátott blokkot *környezetnek* nevezünk” [11]. A L<sup>A</sup>T<sub>E</sub>X-ben számos speciális szövegrész szedésére nyílik lehetőség (felsorolások, idézetek, táblázatok, stb.). Ezen szövegrészek dokumentumba illesztéséhez használhatjuk az ún. környezeteket. Egy példával illusztrálva világosabb lesz. Versek szedésére használható a `verse` környezet. Ez azt jelenti, hogy a beilleszteni kívánt verset a forrásállományban a

```
\begin{verse}
```

```
\end{verse}
```

sorok közé kell beírni. (A környezeten belül a verssorok elválasztására a `\\` parancs szolgál, versszakok közé pedig üres sort kell tenni.)

#### 3.3.2. Igazítás

Alapértelmezés szerint a L<sup>A</sup>T<sub>E</sub>X tömbösített vagy más néven sorkizárt szedést alkalmaz, ami különösen folyamatos szövegeknél szép oldalképet eredményez. A sorszélek margókhöz igazítását a program részint elválasztással, részint a szóközök dinamikus átméretezésével oldja meg.

Vannak esetek – szerencsére igen ritkán –, amikor a L<sup>A</sup>T<sub>E</sub>X nem boldogul egy sortöréssel; ilyenkor megengedi, hogy az adott sor jobbra túllógjon a margón. Ez természetesen elfogadhatatlan, ezért vagy a kilógó szavak lehetséges elválasztási helyeit kell megadni a `\hyphenation{sza-vak lis-tá-ja}` paranccsal, vagy ki kell adni a `\sloppy` parancsot – ennek hatására enyhülnek a sortöréssel kapcsolatos magas követelmények, viszont a nagyobb tűrészhatár miatt nagyobb szóközök keletkeznek, ami nem szép.

Könnyen lehet, hogy a balra igazított szedés jelenti a legelfogadhatóbb megoldást. Ezt javasoljuk olyankor is, amikor valamilyen okból nagyon rövid sorokba tördeljük a mondandónkat. A balra igazítandó szöveget a `flushleft` környezetbe kell elhelyezni. Jobbra igazítani a `flushright`, középre pedig a `center` környezettel lehet.

### 3.3.3. Sortörés, oldaltörés

Röviden átismételve és némileg kibővítve a 3.1.5 szakaszban elmondottakat:

- adott helyen sortörést kicsikarni a `\` vagy a `\newline` paranccsal lehet.
- ha csak sort akarunk törni, de közvetlenül utána új oldalt kezdeni semmiképp, akkor a `\*` lesz jó.
- új oldalon folytatni a szöveget a `\newpage` parancs után lehet.
- szintén sortörést kényszerít ki, illetve tilt meg a `\linebreak[n]` és a `\nolinebreak[n]` parancs. Utóbbiaknál megadhatunk – de nem kötelező, ekkor a zárójeles rész elhagyható – egy négynél nem nagyobb, nem-negatív egész számot. Ha ez a szám 0, 1, 2 vagy 3, akkor a parancsunk lényegében kéréssé szelídül, amit a L<sup>A</sup>T<sub>E</sub>X felülbíráhat, amennyiben a sortörés kevésbé jó, kicsit rossz, rossz vagy nagyon rossz szövegképet eredményezne.
- az előzőekhez teljesen hasonlóan működik a `\pagebreak[n]` és a `\nopagebreak[n]` parancs.

### 3.3.4. Címrendszer, címhierarchia

A 2.7.2. szakaszban áttekintettük röviden, hogy miért és hogyan segítheti egy nagyobb lélegzetű dokumentum logikai felépítésének közvetítését egy jól átgondolt címrendszer. A L<sup>A</sup>T<sub>E</sub>X gyakorlatában ez úgy valósul meg, hogy névvel ellátott *szinteket* használhatunk. A szintek nevei a következők (csökkenő rangsorrendben):

**part** (magyarul: *rész*)

**chapter** (*fejezet*)

**section** (*szakasz*)

**subsection** (*alszakasz* vagy *pont*)

**subsubsection** (*alpont*)

**paragraph** (*paragrafus*)

**subparagraph** (*alparagrafus*)

(Azt is említettük az elméleti részben, hogy a címrendszert nics értelme 4-5 szintnél jobban elmélyíteni. Nem kell tehát a fent említett lehetőségek mindegyikével élni egy dokumentumon belül.)

Cím létrehozásához a szint nevét kell parancs formájában kiadni, közvetlenül utána kapcsos zárójelk közé írva a címet. Például így:

`\subsection{Címrendszer, címhierarchia}`, amivel éppen ennek a pontnak a címét tudjuk megalkotni.

A szintek rendszere általában úgy mutatkozik meg, hogy minél magasabb szinten áll egy cím, annál jobban ki van emelve (vastagabb betűk, nagyobb betűfokozat, nagyobb térköz előtte-utána).

A **book** és a **report** dokumentumosztályokban mindegyik szint használható, a **letter** osztályban egyik sem (egy levelet nem szoktunk fejezetekre darabolni). Az **article** osztályban nincs **chapter** szint. Mindegyik szintparancsnak létezik ún. csillagos változata (pl. `\chapter*{Bevezetés}`), ami létrehozza ugyan az adott szintű címet, de nem ad neki sorszámot.

### 3.3.5. Szóközök, térközök

Szóközökkel a főszövegben viszonylag keveset kell foglalkoznunk: a L<sup>A</sup>T<sub>E</sub>X teszi a dolgát, dinamikusan méretezve egyenletesen osztja el egy adott sorban a szavak közötti üres helyeket (sorkizárt szedés esetén). Természetesen van mód arra is, hogy tetszőleges, de általunk meghatározott mértékű helykihagyásra vegyük rá a rendszert, akár vízszintesen, akár függőlegesen. A `\hspace*{1cm}` parancs például éppen 1 cm *vízszintes térkört* helyez el a szövegben arra a helyre, ahol kiadtuk. Ugyanennek a parancsnak a csillag nélküli változatával kihagyott helyet a T<sub>E</sub>X figyelmen kívül hagyja – „lenyeli” –, ha sor végére vagy sor elejére kerül. Csakúgy, mint a normál szóközt ugyanezekben a helyeken.

Fontos, hogy a kihagyni kívánt hely hossza nem csak cm-ben adható meg. A L<sup>A</sup>T<sub>E</sub>X a mértékegységek ismerete terén sem vall szégyent. A metrikus egységek közül ismeri a mm-t és a cm-t, az angolszász világból az inch-et (jele: in), a számítástechnika nyomdászati egységei közül pedig a pontot (jele: pt és az ún. pica-pontot jelenti, ld. a 2.3.4. szakaszt).

Emellett léteznek a rendszerben az iménti abszolút mértékegységek mellett relatív mértékek is, amelyek konkrét mérete az adott helyen használt betű fokozatától és változatától függ. Ilyenek az `ex` és az `em`. Előbbi az „x” betű magasságával, utóbbi nagyjából a betűfokozattal egyezik meg. Említhetnénk még a rugalmas hossz méreteket is, melyek értékét a T<sub>E</sub>X bizonyos – általunk megadott – határok között változtathatja a minél szebb tördelés érdekében. Részletek erről a [11] 84. oldalán.

*Függőleges térkört* létrehozni a `\vspace*{hossz}` parancssal lehet, mégpedig éppen „hossz”-nyit. Lényeges különbség a `\hspace*{hossz}`-hoz képest, hogy a függőleges helykihagyás nem feltétlenül a parancs kiadásának helyén történik, hanem ott, ahol a T<sub>E</sub>X a sort eltöri, vagy ahol új bekezdés kezdődik (általánosabban: ahol a T<sub>E</sub>X ún. függőleges módba kerül). Itt is használhatjuk viszont a csillag nélküli változatot, ekkor az éppen új oldal tetejére eső függőleges térköz nem fog megjelenni.

### 3.3.6. Felsorolás, számozás

(Ehhez a szakaszhoz óravázlat és prezentáció található a szakdolgozat 2. számú mellékletében.) Dokumentumainkban előfordulhatnak – és nem ritkán elő is fordulnak – lista jellegű szövegrészek. Ezeknek három típusát különböztetjük meg: egyszerű felsorolás, sorszámozott lista, leíró lista (más néven szótárszerű felsorolás).

**Egyszerű felsorolás** létrehozására használjuk az `itemize` környezetet. A `\begin{itemize}` és az `\end{itemize}` sorok közé kell beírni a lista elemeit, mindegyiket az `\item` paranccsal kezdve. Bármelyik listaelem maga is tartalmazhat listát, ezek tehát egymásba ágyazhatók, de nem korlátlanul: legfeljebb négy mélységig. A dokumentum minden szintjének saját felsorolásjele van, ami a listaelemeket bevezeti. Megváltoztatható, ha az elemeket `\item[sajátjel]` formában írjuk be a forrásállományba.

**Sorszámozott lista** keletkezik az `enumerate` környezetben. Itt is van lehetőség egymásba ágyazásra, a mélységi korlát megegyezik a felsorolásnál elmondottal. Az első szinten arab számok, a másodikokon kisbetűk zárójelben, a harmadikon kis római számok, a negyedik szinten pedig nagybetűk jelölik a listaelemeket.

**Szótárszerű listák** hozhatók létre a `description` környezet segítségével. A címszó megadására az `\item` parancs után közvetlenül, szögletes zárójel között van lehetőség, valahogy így: `\item[címszó]`. Ebben a környezetben a rendszer hat mélységig enged meg egymásba ágyazást. A címszavakat a L<sup>A</sup>T<sub>E</sub>X félkövér betűváltozattal fogja szedni, ami ebben az esetben a szövegközi kiemelésnek megengedhető, sőt, javasolt módozata.

### 3.3.7. Képek és táblázatok

Táblázatokat készíteni L<sup>A</sup>T<sub>E</sub>X környezetben nem tartozik a legegyszerűbb feladatok közé, körülményesnek is tűnhet kissé, ám van néhány pozitívum, ami miatt talán mégis megéri a fáradságot. Nagyon szép végeredmény születhet,

az elkészült táblázatra könnyű hivatkozni, és ha sok táblázatot tartalmaz a dokumentumunk, akkor gyerekjáték hozzájuk jegyzéket készíteni.

Tekintsünk először egy példát, amelyben a középszintű érettségi százalékos ponthatárait foglaljuk össze. Ehhez az alábbi kódot kellett elhelyezni a

3.1. táblázat. Érettségi ponthatárok

eredmény (%)	osztályzat
80–100	jeles
60–79	jó
40–59	közepes
20–39	elégséges
0–19	elégtelen

forrásállományban:

```
\begin{table}[!h]
\caption{Érettségi ponthatárok}
\center
\begin{tabular}{|c|l|}
\hline
eredmény (%) & osztályzat \\
\hline
80--100 & jeles\\
\hline
60--79 & jó\\
\hline
40--59 & közepes\\
\hline
20--39 & elégséges\\
\hline
0--19 & elégtelen\\
\hline
\end{tabular}
```

Nézzük sorról sorra! A `table` környezet gondoskodik arról, hogy táblázatunkból ún. úszó objektum legyen, ami nagy vonalakban azt jelenti, hogy a T<sub>E</sub>X a dokumentum tördelésekor a legmegfelelőbb helyre fogja elhelyezni. Ez a „legmegfelelőbb hely” lehet a szövegnek az a része, ahol a táblázatot létrehozzuk (`h`), lehet a lap teteje (`t`), a lap alja (`b`), vagy akár egy külön lap (`p`) is. A `\begin{table}` után szögletes zárójel között a megfelelő betűk felsorolásával megadhatjuk a T<sub>E</sub>X-nek, hogy milyen sorrendben próbálkozzon az ideális elhelyezéssel. Ha nem adunk meg semmit, akkor a `[tbp]` lesz érvényben, tehát elsősorban az oldal tetején, aztán az oldal alján, végül külön lapon igyekszik elhelyezni. Lehetünk szigorúbbak is: a példában a `[!h]` azt jelenti, hogy ezt a táblázatot mindenképpen oda kell tenni, ahol azt létrehoztuk.

A következő sorban a `\caption` parancs révén adhatunk címet az objektumnak.

A `\center` utasításból tudja meg a L<sup>A</sup>T<sub>E</sub>X, hogy középre kell igazítania.

Magát a táblázatot a `tabular` környezettel hozzuk létre. A környezeten belüli első sorban az derül ki (a `{|c|l|}` kódrendszerben), hogy ez egy kétoszlopos táblázat lesz, a középre igazított („c”) első oszlop előtt, a két oszlop között és a balra igazított („l”) második oszlop után függőleges vonallal. A függőleges vonalak helyét – amint az előbbiekből bizonyára már kiderült – a `|` jellel adhatjuk meg. Vízszintes vonalat a `\hline` paranccsal húzhatunk, mégpedig azon sor fölé, amelyik elé beírjuk. Példánkban az első sor fölé és az utolsó sor alá is került ilyen. A táblázat celláinak tartalmát soronként adjuk meg, a soron belüli cellatartalmakat az `&` jellel választva el egymástól, a sor végét pedig a `\\` jelekkel jelezve. A végén még természetesen egy `\end{tabular}`, valamint egy `\end{table}`, és készen is vagyunk! (A `table` környezettel mintegy beburkoltuk a `tabular` környezetet.)

A L<sup>A</sup>T<sub>E</sub>X-es táblázatkészítés persze ennél lényegesen sokrétűbb; egy tömör összefoglaló olvasható róla az Egy nem túl rövid bevezető...-ben [7], egy mindenre kiterjedő, részletes leírás pedig a L<sup>A</sup>T<sub>E</sub>X kézikönyvben [11].

Képeket varázsolni a dokumentumba lényegesen egyszerűbb – feltéve, hogy azok EPS (*Encapsulated Post Script*) formátumúak. Képeink többsége nem ilyen. Sebaj, az iskolai gépek tekintélyes hányadán – nálunk például mindegyiken – fellelhető a GIMP nevű, méltán népszerű képszerkesztő és

-konvertáló szabad szoftver, amelynek segítségével néhány kattintással `.eps`-sé alakulhatnak hétköznapiabb formátumú (`.jpg`, `.gif`, `.png`, `.bmp`, stb. – utóbbi már nem fájlkiterjesztés...) képeink.

A dokumentumba illesztéshez használjuk ki a T<sub>E</sub>XnicCenter szolgáltatásait. Kép beszúrására módunk van menüből (Insert → Picture...), billentyűkombinációval (Ctrl+Alt+G vagy ikonnal (kaktusz és napocska van rajta). Mindhárom út egy Insert Figure nevű párbeszédpanelhez vezet, amelyen kitallózhatjuk a beillesztendő képfájlt, rendelkezhetünk a középre igazításról, az objektum „úsztatásáról” tördeléskor – utóbbiak alapbeállítások a panelen –, megadhatjuk a képaláírás szövegét (Caption), és elláthatjuk a képet olyan címkével, amelynek segítségével a dokumentumban bárhol hivatkozhatunk rá (Label). Még mindig ugyanígy beállíthatjuk a kép pozíciójára vonatkozó kéréseinket ugyanúgy, mint a táblázatoknál, mindössze annyi különbséggel, hogy itt jelölőnégyzetek segítségével válogathatunk. Az alapértelmezés képeknél is a [tbp]. Ha mindent beállítottunk és kattintottunk az OK gombra, akkor meglátjuk a forrásállományban azt a kódrészletet, amelyet begépelve is eljárhattunk volna (benne a `figure` környezettel, a `\caption` és `\label` parancsokkal).

### 3.3.8. Idézetek, versek

Szövegeinkben a hosszabb-rövidebb kiemelt idézeteknek is illik szépen kinézni. Nem kell elvesznünk a részletekben, a L<sup>A</sup>T<sub>E</sub>X kész megoldásokat szállít ezekre az esetekre. Teszi mindezt kellőképpen árnyaltan: különböző környezetek szolgálnak a rövid (általában egy mondatos) és a hosszú (több bekezdésen át tartó) idézetek szedésére. Előbbihez a `quote`, utóbbihoz a `quotation` környezet lesz alkalmas. Ha verset szeretnénk beilleszteni a dokumentumba, akkor járjunk el úgy, ahogy a 3.3.1 pontban olvasható (ott éppen a `\verse` környezettel példálóztunk).

## 3.4. Matematika, képletek

Ahol a T<sub>E</sub>X és a L<sup>A</sup>T<sub>E</sub>X elvitathatatlanul nagyon erős – még önmagához képest is –, az a matematikai tárgyú szövegek, képletek, formulák szedése, szerkesztése. A tételek, bizonyítások, definíciók egyrészt egységes és szép megjelenésűek, másrészt – ami tartalmilag nagyon fontos – hivatkozhatók lesznek. A képletek, egyenletek pedig tipográfiailag kifogástalan kinézetet nyernek. Mindezek miatt a T<sub>E</sub>X és a L<sup>A</sup>T<sub>E</sub>X a műszaki-tudományos világban töretlenül népszerű, a matematikusok körében pedig egyenesen megkerülhetetlen.

### 3.4.1. Definíciók, tételek, bizonyítások

Megjelenésüknek a dokumentumban mindvégig egységesnek, áttekinthetőnek, világosnak kell lenni. Megoldást az ún. tételszerű környezetek jelentenek, melyeket a preambulumban kell definiálni. Formailag ez a következőket jelenti például:

```
\newtheorem{ttl}{tétel}
```

elsőként az újonnan definiált környezet neve, aztán a megjelenítendő szöveg

```
\newtheorem*{biz}{\emph{Bizonyítás}:}
```

a \*-os változat itt is azt eredményezi majd, hogy a bizonyítás nem kap sorszámot

```
\newtheorem{fdt}{feladat}[section]
```

a feladatok sorszáma szakaszonként induljon újra

```
\newtheorem{pld}{példa}[section]
```

Ezek után a dokumentumban már gond nélkül használhatjuk a definiált környezeteket, például így:

```
\begin{ttl}
```

Derékszögű háromszög befogóira emelt négyzetek területeinek összege egyenlő az átfogóra emelt négyzet területével.

```
(Pitagorasz tétele)
\end{ttl}
\begin{biz}
a befogótétel alapján triviális\dots
\end{biz}
```

### 3.4.2. Matematikai formulák leírása

Három matematikai módot különböztetünk meg egymástól:

- soron belüli matematikai mód (dollárjelek közé kerül a matematikai kifejezés, vagy a `math` környezetbe)
- külön sorba kerülő, de nem számozott egyenlet (`displaymath` környezetbe, vagy a `\[` és a `\]` jelpárok közé írjuk a matematikai kifejezést)
- számozott, külön sorba írt egyenlet (`equation` környezet). Az utóbbi címkézhető (`\label{cimke}`-vel) és hivatkozható.

Mindhárom eset kényelmesen kezelhető a `TEXnicCenter Insert`  $\longrightarrow$  `Formulas` menüjéből is. Szövegközi képlet például a  $\sin^2 x + \cos^2 x = 1$  („trigonometrikus Pitagorasz-tétel” fantázianevét) összefüggés, melyhez a `\$sin^2x+cos^2x=1\$` kódot kellett beírni.

Kiemelt – külön sorban, előtte-utána térközzel megjelenő – formulánk legyen mondjuk a másodfokú egyenlet megoldóképlete:

$$x_{1;2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

Az ennek megjelenítéséhez szükséges kód első látásra nem feltétlenül bizalomgerjesztő:

```
\[ x_{1;2}=\frac{-b\pm\sqrt{b^2-4ac}}{2a}. \], de ha jobban meg-
```

vizsgáljuk, látni fogjuk, hogy sokkal logikusabb, mint amennyire bonyolult. (Kifejezetten elegáns például a „pluszmínusz” jel bevitelére szolgáló `\pm` parancs.)

Grafikus felhasználói környezeteken nevelkedetteknek – és ma már minden középiskolás ilyen – jó hír lehet viszont, hogy az iménti kódból a képletben

szereplő betűkön és számokon kívül szinte semmit nem kellett billentyűzetről bevinni: a T<sub>E</sub>XnicCenter eszköztárán mindenhez van megfelelő ikoncsoport (csak legyen bekapcsolva a `Toggle Math Bar` nevű ikon). Ezek segítségével tetszőlegesen bonyolult formulák is játszani könnyedséggel beszerkeszthetők – és ez néhány perc gyakorlás után egyáltalán nem tűnik majd akkora túlzásnak, mint esetleg most.

A (3.1) exponenciális egyenlet egy kiemelt, sorszámozott formula:

$$x^{\lg \sin x} + x^{\lg \cos x} = 2 \quad (3.1)$$

Egy `equation` környezetbe került, a következő formában:

```
\begin{equation}
\label{eq:pelda}
x^{\lg \sin x} + x^{\lg \cos x} = 2
\end{equation}.
```

A hivatkozás az `Aref({eq:pelda})` paranccsal történt így:

„`Aref({eq:pelda})` exponenciális egyenlet...” A L<sup>A</sup>T<sub>E</sub>X a magyar nyelvet támogató csomagja segítségével ilyenkor kiírja a helyesen illesztett határozott névelőt is.

A T<sub>E</sub>X-es képletszerkesztés kulcsa egyszerűen a rutin. Némi gyakorlással valóban virtuóz dolgokra lehetünk képesek. Remek terep ehhez az *Art Of Problem Solving* [14] című weboldal, ahol a L<sup>A</sup>T<sub>E</sub>X → TeXer menüpontban próbálgathatjuk csak a formulák szerkesztését úgy, hogy az eredményt gombnyomásra ott helyben megtekinthetjük (és ha nagyon tetszetős, képként akár el is menthetjük). Ugyanezen a honlapon, az előzőtől két kattintásnyira pedig gyakorlatilag az összes szimbólum, függvény, parancs felsorolását is megtaláljuk.

Fontos apróságok, amikre figyelniük kell képletek szerkesztésekor:

- A mondatvégi írásjelet szövegközi képlet esetén a képleten kívül, kiemelt formulánál ellenben a képleten belül helyezzük el.
- Tizedes elválasztásra a pont (.) szolgál, mivel a program amerikai eredetű. Mi nyilván tizedesvesszőt szeretnénk látni, a vesszőt viszont a

felsorolás jelének fogja értelmezni a L<sup>A</sup>T<sub>E</sub>X, így a  $\pi \approx 3,14$  helyett ezt fogjuk látni:  $\pi \approx 3,14$ . Ez ebben a formában teljesen rossz. A megoldás némi negatív térköz – igen, ilyen is van – betoldása. A negatív térköz parancsa a `\!`. Az előbbi példát szépen megjelenítő kódrészlet tehát: `\$ \pi \approx 3, \! 14 \$`.

- Képletbe szöveget beírni az `\mbox{szöveg}` parancs tud. Ezt inkább olyankor használjuk, ha a formula mellé írunk valamit, mondjuk magyarázatként. Példa:

$$(-1)^n = 1, \text{ ha } n \text{ páros.}$$

Így készült: `\[ (-1)^n=1, \mbox{ha } n \text{ páros} . \]` Az `\mbox` parancs nem alkalmas viszont arra, hogy a képletben szerepeltessünk szöveget. Nézzük meg a gömb térfogatára vonatkozó összefüggést először `\mbox`-szal:

$$V_{\text{gömb}} = \frac{4}{3} \cdot \pi \cdot r^3$$

Nem szép. Jó megoldást a problémára a `\text{szöveg}` parancs jelent (be kell hozzá tölteni az `amsmath` csomagot, úgyhogy irány a preambulum). Az előző képlet kódja helyesen tehát:

`\[V_{\text{gömb}}=\frac{4}{3} \cdot \pi \cdot r^3\]`, ennek eredménye pedig:

$$V_{\text{gömb}} = \frac{4}{3} \cdot \pi \cdot r^3.$$

A különbség jól látható.

- Kombinatorikában nem nélkülözhetjük az „ $n$  alatt a  $k$ ” típusú kifejezéseket. Ilyeneket az `\$n \choose k\$` kóddal hívhatunk elő.
- Alsó indexbe a `_{valami}`, felsőbe pedig a `^{\text{valami}}` formában lehet írni.

Ez a szakasz legfeljebb egy gyors bepillantást engedett a L<sup>A</sup>T<sub>E</sub>X matematikai világába; a teljesség igényével fellépő leírás akár önálló kiadvány tárgya is lehetne. Érdeklődőknek szakirodalom gyanánt ugyanaz ajánlható, mint a 3.3.7. szakasz végén.

## 3.5. Jegyzetek, jegyzékek

Mondanivalónk lényegi részét a folyószöveg tartalmazza, amit magyarázó jegyzetek segítségével próbálhatunk meg még érthetőbbé tenni, illetve plusz információkkal kiegészíteni. Egy néhány oldalasnál hosszabb dokumentumnak ugyancsak fontos tartozéka a tartalomjegyzék, amit a kiadvány elején vagy végén szokás elhelyezni; komolyabb dolgozatok, pályamunkák esetén pedig elvárható, hogy a felhasznált forrásokat felsoroló irodalomjegyzék is készüljön.

### 3.5.1. Tartalomjegyzék létrehozása

Tartalomjegyzéket végtelenül egyszerűen készíthetünk, mindössze ki kell adni a `\tableofcontents` parancsot. Első fordításkor ennek hatására létrejön egy `.toc` kiterjesztésű fájl – amivel nekünk érdemben nincs dolgunk –, másodszorra ugyanez beszerkesztődik a kimeneti fájlba (a `.dvi`-be vagy a `.pdf`-be alapvetően). Helyes eredmény eléréséhez tehát *kétszer kell lefordítani* a forrásállományt.

Lássunk most két fontos lehetőséget a tartalomjegyzék finomhangolására. A 3.3.4. pontban láttuk, hogy a címrendszer létrehozásakor van módunk sorszám nélküli rész-, fejezet-, stb. címeket létrehozni a megfelelő parancsok csillagos változatával. Csakhogy amit így hozunk létre, az alapesetben nem kerül be a tartalomjegyzékbe. Ha mégis ott szeretnénk látni, akkor a következőképpen járhatunk el – például:

```
\chapter*{Bevezetés}
\addcontentsline{toc}{chapter}{Bevezetés}
```

A három paraméterrel a következőket adhatjuk meg: *a)* hová szeretnénk elhelyezni (itt: `toc`, vagyis a tartalomjegyzékbe) *b)* milyen címszinten kerüljön be (itt: `chapter`, azaz fejezetcím legyen), illetve *c)* milyen szöveggel jelenjen meg ott (túl sok értelme persze nem lenne a „Bevezetés”-t máshogyan fogalmazni).

A másik lényeges apróság, hogy a tartalomjegyzék mélysége is állítható. Ha például azt szeretnénk, hogy csak a részek és a szakaszok címei kerülje-

nek be, akkor a `\tableofcontents` paranccsal a `setcounter{tocdepth}{1}` parancsot is ki kell adnunk. Amennyiben a második paraméter értéke nem egy, hanem kettő, akkor az alszakaszok címei is megjelennek a tartalomjegyzékünkben. (És még egyszer: kétszer fordítani!)

### 3.5.2. Lábjegyzet, széljegyzet, végjegyzet

A L<sup>A</sup>T<sub>E</sub>X-ben háromféle jegyzet készíthető könnyedén:

**Lábjegyzet**et a `\footnote{A lábjegyzet szövege}` paranccsal hozunk létre.

Közvetlenül – szóköz nélkül – írjuk az után a szó után, amire vonatkozik. Ha nem egy szót, hanem egy vagy több mondatot érint, akkor pedig rögtön a mondatvégi írásjelet követően helyezzük el.

**Széljegyzet** kerülhet az oldal margójára a `\marginpar{szöveg}` parancs segítségével. Rövid fogalommagyarázatokhoz hasznos és hatékony formai megoldás. Középiskolai tankönyvekben gyakran találkozhatunk vele.

**Végjegyzet** létrehozására is van lehetőség – ez a lábjegyzetnek egy olyan speciális formája, ami a fejezet vagy az egész mű végére kerül –, de ezt a L<sup>A</sup>T<sub>E</sub>X alapkészlet már nem tartalmazza. Be kell tölteni hozzá az `endnotes` nevű csomagot (ami egyszerűen annyit jelent, hogy a forrásállomány preambulumaiba be kell írni a `\usepackage{endnotes}` sort). Így már működni fog a végjegyzet létrehozására szolgáló `endnote{A végjegyzet szövege}` parancs.

### 3.5.3. Irodalomjegyzék beillesztése

Ha pl. iskolai pályamunkát követünk el, akkor egészen biztos, hogy forrásokból dolgozunk. Ezeket a forrásokat a dolgozat végén fel kell sorolnunk, mégpedig úgy, hogy e felsorolás megfeleljen bizonyos formai előírásoknak. Ezekre itt nem térünk ki részletesen, de JENEY GÁBOR: *Hogyan néz ki egy igényes dokumentum?* [4] című jegyzetében utána lehet nézni.

A L<sup>A</sup>T<sub>E</sub>X-ben létrehozott irodalomjegyzék tételeire hivatkozhatunk is a folyószövegben bárhol. Egyszerűbb irodalomjegyzékek készítését a `thebibliography`

környezet támogatja. A kezdősor `\begin{thebibliography}{9}` vagy `\begin{thebibliography}{99}` legyen. Előbbi akkor, ha tíznél kevesebb, utóbbi akkor, ha száznál kevesebb, de kétszámjegyű tételből áll a címlistánk. A környezet belsejében minden tételt a `\bibitem{cimke}` vezessen be, ezt követi majd a szerző, a mű címe, stb. Az irodalomjegyzékben szereplő művekre, weboldalakra hivatkozni úgy tudunk, hogy a hivatkozás helyén kiadjuk a `\cite{cimke}` parancsot. Pár sorral feljebb is így történt.

## 3.6. A dokumentum elektronikus megjelenítése

Mindennapi működésünk során keletkező dokumentumaink között lehetnek olyanok, amelyeket elektronikus formában (is) szeretnénk publikálni. Alapvetően két út áll előttünk: készíthetünk html-dokumentumot, amely aztán akár egy weboldalnak is része/egésze lehet, vagy gyárthatunk .pdf-fájlt, amit letölthetővé, illetve egyéb más módon elérhetővé teszünk. Előbbi esetben szükségünk lesz a `latex2html` nevű programra, melynek használatáról [15]-ből tájékozódhatunk. Utóbbihoz viszont elegendő a forrásállomány lefordításakor a `LaTeX=>PDF` lehetőséget választani. Egyszerűbb felépítésű, grafikus elemeket nem tartalmazó dokumentumoknál legalábbis. Speciális esete a .pdf állományok létrehozásának a prezentációkészítés, melyről külön is érdemes szólni.

### 3.6.1. Prezentációkészítés PDF-alapokon

Ha prezentációt készítünk középiskolai használatra, általában nem PDF formátumban gondolkodunk. Ami miatt mégis érdemes letérni a „hagyományos” útról – ez jobbára a tananyagban is szereplő PowerPoint nevű szoftver használatát jelenti –, az egyrészt a .pdf fájlok nagyfokú hordozhatósága, másrészt a L<sup>A</sup>T<sub>E</sub>X-től megszokott verhetetlen színvonalú megjelenés.

Több csomag is készült az idők során ennek támogatására, így pl. egy tanórai kiselőadást támogató prezentációt elkészíteni viszonylag egyszerűen lehet. Lássuk, hogyan! Munkánkhoz a `beamer` nevű csomagot fogjuk használni. Dokumentumosztályként is ezt kell megadni, tehát a `preambulum` a

`\documentclass{beamer}` paranccsal fog kezdődni. A prezentáció építőelemei a *frame*-ek. A `frame` környezet jelenti azokat a diákat, amiket látunk, amelyek között tudunk a prezentáció vetítésekor léptetni, ezek tartalmazzák a szöveget, felsorolásokat, ábrákat, tehát magát a dokumentumot. Frame-eket a `frame` környezettel készíthetünk a következő módon:

```
\begin{frame}
Az adott dia tartalmát írjuk ide
\end{frame}
```

A szokásos címdia elkészítése például ennyit jelent mindössze:

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

A `\titlepage` parancsnak persze csak akkor van értelme, ha korábban már megadtuk az előadás címét, szerzőjét (itt inkább az előadó nevét), dátumát rendre a `\title`, a `\author`, illetve a `\date` parancsok argumentumaiban.

Ha tartalomjegyzéket is szeretnénk vetíteni az előadás elején, akkor természetesen logikailag tagolnunk kell a mondandónkat a korábban már megismert módon (`\section`, `\subsection`, stb.). Ezeket a címszinteket a `frame`-eken kívül adjuk meg. A tartalomjegyzék kapjon egy külön diát (`frame-et`), a következőképpen:

```
\begin{frame}
Tartalomjegyzék:
\tableofcontents[pausesections]
\end{frame}
```

Ellentétben egy „sima” L<sup>A</sup>T<sub>E</sub>X-ben készült dokumentummal, itt a „Tartalomjegyzék” szót mi írjuk be a forrásállomány szövegébe, ha láttatni akarjuk. A `[pausesections]` – nem kötelező – opció segítségével pedig azt érjük el, hogy a szakaszok címei nem egyszerre, hanem egymás után fognak megjelenni vetítéskor.

Prezentációnk megjelenési stílusát részenként vagy egészében az alábbi parancsokkal állíthatjuk be:

- `\usetheme{Témanév}`: ezzel a bemutató általános összképét határozzuk meg; a különböző témákat városokról nevezték el (Frankfurt, Berlin, Madrid, PaloAlto, CambridgeUS, Szeged, stb.), érdemes kísérletezni.
- `\usecolortheme{témanév}`: a prezentáció színvilágára leszünk hatással, ha kiadjuk.
- `\useinnertheme{témanév}`: elsősorban a felsorolások, listák kinézetét fogja befolyásolni.
- `\useoutertheme{témanév}`: azt mondjuk meg általa, hogyan nézzenek ki a fejlécek, címek, lábjegyzetek.
- `\usefonttheme{témanév}`: a felhasznált betűtípus beállítására szolgál.

Fentiek megadása nem kötelező, kezdetben hagyjunk mindent az alapértelmezés szerinti beállításban, így is rendkívül mutatós, ugyanakkor ízléses végeredményt kapunk.

A L<sup>A</sup>T<sub>E</sub>X-ben általában is előforduló környezeteket (definíciók, tételek, bizonyítások, képletek, egyenletek, idézetek, versek, stb.) a `beamer` dokumentumosztályban gond nélkül használhatjuk.

Külön érdemes szólni a felsorolásokról. Általános igény, hogy előadásunk során a listák elemei ne egyszerre, hanem egymás után jelenjenek meg. A `\pause` parancs éppen ebben nyújt segítséget. Egyszerűen a megfelelő listaelemek közé kell elhelyezni.

A szövegközi kiemelésnek van egy nyomatékosabb formája `beamer` prezentációkban, ez pedig az `\alert{kiemelendő szöveg}` parancs.

A végeredmény – vagyis a `.pdf` fájl – elkészítéséhez a TeXnicCenter eszköztárán a szokásos `LaTeX=>DVI` helyett ezúttal válasszuk ki a legördíthető listából a `LaTeX=>PS=>PDF` opciót. És a biztonság – meg a helyes eredmény – kedvéért fordítsuk le kétszer a forrásállományt. Az elkészült PDF formátumú dokumentumot az `Adobe Reader` nevű programmal nyissuk meg.

Az egyszerre egyszerű és sokoldalú `beamer` csomagról részletes és élvezetes leírás található a `beamer` felhasználók kalauzában [9], nagyon rövid összefoglaló pedig a Matematikai Prezentációkészítés című wikilapon [13].

# Összegzés

Szakedolgozatomban arra tettem kísérletet, hogy egy középiskolai – azon belül is elsősorban gimnáziumi – tananyagot hozzak létre, amelynek segítségével a tanulók már a felsőoktatásba kerülésük előtt megismerkedhetnek a tipográfia alapvető fogalmaival, szabályaival.

A magyar közoktatásban jelenleg érvényben lévő kerettantervnek az informatikai műveltségterületre vonatkozó fejezeteit tanulmányozva az derült ki, hogy szövegszerkesztő program kezelési ismereteket mindenképpen tanulnak ugyan a gimnazisták – a 9. évfolyamon –, de az erre fordítható órakeret legfeljebb egy WYSIWYG szövegszerkesztő használatának elsajátításához elegendő, alapfokon. Mindennapi tanári gyakorlatomban közvetlenül is van módomban megtapasztalni, hogy ez nem ritkán kritikán aluli kinézetű dokumentumokat képes eredményezni.

Témaválasztásomban fontos tényező volt, hogy szeretném a közoktatásból gyakorlatilag teljes mértékben hiányzó tipográfiai alapismereteket az általános műveltség részévé tenni. Az ehhez választott és feldolgozott források azt igazolták számomra, hogy bár a témakör ismeretanyaga rendkívül gazdag, létezik olyan válogatás, amely az alapvető tudnivalókat tartalmazza.

Fontos volt továbbá tisztázni, hogy ebben a válogatásban – és ezzel együtt ebben a dolgozatban is – csak a tipográfia kiadványtervezési vonatkozásai szerepelhetnek, és el kell tekinteni a betűtervezéssel kapcsolatos tudnivalók tárgyalásától. Például azért is, mert utóbbi – meglátásom szerint – alapfokon nem művelhető.

Nem titkolt célja volt jelen dolgozatnak az is, hogy a dokumentumok igényes megjelenésével kapcsolatos tudnivalókat – nem feltétlenül informatika szakos – tanárok számára is hozzáférhetővé tegye.

Írtam két olyan kísérletről ([2] és [5]), melyek azt igazolják, hogy a diákok elé kerülő és tanulásra szánt szövegek tipográfiája és a szöveg értő elolvasása, illetve megtanulása között összefüggés mutatható ki. Tanári felelősségünk van tehát abban, hogy az általunk készített dokumentumok tipográfiailag kifogástalanul kerüljenek tanítványaink elé. Jelen szakdolgozat első fejezete fentieket szem előtt tartva készült.

Terjedelmi okokból meg kellett hozni azt a döntést, amely alapján a többszörös szedésre, a képek elhelyezésére és a modulhálós oldaltervezésre vonatkozó tudnivalók nem kerülhettek be a fejezetbe. A későbbiek folyamán – amikor is a dolgozat második és harmadik fejezetéből önálló jegyzet készül – a tananyag ezekkel bővíthet.

A szakdolgozat megírása közben derült ki az is, hogy az alapvető tipográfiai tudnivalók nem csupán szakköri keretben tárgyalhatók, de lehetséges ezeket speciális esetben a kötelező órakeretbe is beilleszteni. Ez a speciális eset pedig a nyelvi előkészítő gimnáziumi osztályok ún. „nulladik” évfolyama.

Ezen az évfolyamon egyrészt meglehetősen magas lehet az informatika órák száma (heti öt óra általában), másrészt sok helyen az Európai Számítógéphasználati Jogosítvány (ECDL) megszerzésére való felkészítést építik be a tantervbe, ily módon a szövegszerkesztő program kezelési ismeretek tanítása önálló modult kap a tanulmányok során. Harmadsorban pedig bátran lehet alapozni a tanulók általános iskolából hozott előismereteire az Operációs rendszerek, valamint az Információ és kommunikáció című modulok terén, és lefaragni az ezekre betervezett összóraszámából. Fentiekből következik, hogy a szövegszerkesztési ismereteket tartalmazó tananyag részbe becsempészhetők a tipográfiai tudnivalók, még a – jellemzően MS Word – szövegszerkesztő szoftver kezelésének tanítása elé.

Gimnáziumban programozási ismereteket tanulni kizárólag azoknak kell, akik emelt szintű érettségét szeretnék tenni informatikából. Ők voltak a szakdolgozat második fejezetének egyik potenciális célcsoportja, mivel a L<sup>A</sup>T<sub>E</sub>X használata részint megköveteli, részint erősíti azt a fajta strukturális gondolkodásmódot, amire a programozásban is szükség van.

A T<sub>E</sub>X-re épülő szöveg- és kiadványszerkesztés oktatása középiskolában

viszont már semmilyen „trükközéssel” nem csempészhető be a kötelező tananyagba, így az erről szóló fejezet mindenképpen fakultatív – például szakköri – feldolgozást kíván. Vannak ennek a formának előnyei is: egyrészt az ismeretanyag összeállításában és a feldolgozás módjában az oktató az átlagosnál is jóval nagyobb szabadságot élvez, másrészt pedig a kurzusban részt vevő tanulók motiváltsága eleve adott, így a tanítás hatékonysága lehet komolyabb mértékű.

A fejezet anyagának végiggondolásakor a bőség zavara és ebből fakadóan a zavar bősége váltogatták egymást. A végeredmény már tartalmazza mindazon kompromisszumokat, amelyeket úgy a terjedelmi, mint az érthetőségi korlátok miatt meg kellett kötni. Akár csak eggyel magasabb szintű L<sup>A</sup>T<sub>E</sub>X-kurzus sem képzelhető el például a `babel`, a `[T1]{fontenc}` vagy az `inputenc` csomagok részletesebb ismertetése – egyáltalán: ismertetése – nélkül, míg itt erre nem került sor.

Hiányérzetre adhat okot az is, hogy a L<sup>A</sup>T<sub>E</sub>X grafikai lehetőségeinek bemutatása lényegében szintén kimaradt, és a táblázatok létrehozásáról is meglehetősen szűkszavúan szól a szöveg. Utóbbit azért valamelyest indokolja, hogy csak erről szóló szakdolgozat is született már.

Összességében az anyag kapcsán némi párhuzam fedezhető fel a gimnáziumok matematika fakultációin oktatók emelt szintű analízis témakörrel. Tanítunk ugyan differenciál- és integrálszámítást, de csak olyan mértékben és mélységben, hogy az alapvető összefüggésekkel, szabályokkal tisztában legyenek, tudjanak egyszerűbb gyakorlati problémákat megoldani ezekkel az eszközökkel, és végül, de nem utolsósorban a felsőoktatásban szembesülve a témakörrel ne kelljen az abszolút nulláról indulniuk.

Amivel az anyagot a későbbiekben feltétlenül bővíteni fogom: *a*) érdekességek, különleges szedési megoldások bemutatása a L<sup>A</sup>T<sub>E</sub>X határtalan képességeinek demonstrálására (sajátos alakú szövegfoltok, sakkállások, kották létrehozása), illetve *b*) példatár és feladatgyűjtemény megalkotása a matematikai formulák szerkesztésének gyakoroltatásához (az emelt szintű matematika érettségi szóbeli tételsorához mellékelt képlettár elkészítése – kooperatív módon, csoportmunkában – L<sup>A</sup>T<sub>E</sub>X-ben)

Végeredményben a szakdolgozat tervezésekor megfogalmazott rövid távú

célokat (nevezetesen, hogy jöjjön létre olyan középiskolai fakultatív tananyag, amely a legalapvetőbb tipográfiai ismeretek tanítását és elsajátítását segíti, és amely segítséget nyújt a T<sub>E</sub>X és a L<sup>A</sup>T<sub>E</sub>X megismertetéséhez és megismeréséhez egyaránt) lényegében sikerült elérni. A távlatosabb megfontolások – a tipográfia már a középfokú oktatásban is váljon az ismeretanyag, de még inkább az általános műveltség részévé – realizálása nyilvánvalóan hosszabb folyamat eredménye lehet majd. A magam részéről jelen dolgozattal és az ezt követő informatika tanári munkával a folyamatot katalizálni igyekszem.

# Irodalomjegyzék

- [1] BUJDOSÓ GYÖNGYI, *Néhány alaptanács szövegek szerkesztéséhez, 1. rész*, DE IK, Debrecen, 2005.
- [2] GÓSY MÁRIA, „A szövegértő olvasás” *Anyanyelv-pedagógia*, Magyar Nyelvtudományi Társaság Magyartanári Tagozata  
<http://www.anyanyelv-pedagogia.hu/cikkek.php?id=25>  
2008. július 20.
- [3] GYURGYÁK JÁNOS, *Szerkesztők és szerzők kézikönyve*, Osiris, Budapest, 1996.
- [4] JENEY GÁBOR, *Hogyan néz ki egy igényes dokumentum?*,  
<http://kutfo.hit.bme.hu/oktatas/kinezet.pdf>  
2008. július 20.
- [5] KARAP GERGŐ, *Tipográfiai módszerek alkalmazásának hatása az előhívási teljesítményre*,  
<http://pszicho.btk.ppke.hu/diakelet/munkaink/kgmmaltpszi.doc>  
2008. augusztus 11.
- [6] NAGY BENCE, „Kényszer szülte tökéletesség”, *Magyar Grafika*, 2005. Expo különszám, 87–90. oldal
- [7] T. OETIKER–H. PARTL–I. HYNA–E. SCHLEGL, *Egy nem túl rövid bevezető a L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> használatába*

- [8] SZABÓ ZÉNÓ LÁSZLÓ, *Tipográfia a weben*  
<http://edutech.elte.hu/jegyzet/tipografia/>  
2008. július 20.
- [9] TILL TANTAU, *User Guide to the Beamer Class, Version 3.07*  
[www.ctan.org/tex-archive/macros/latex/contrib/beamer/doc/  
beameruserguide.pdf](http://www.ctan.org/tex-archive/macros/latex/contrib/beamer/doc/beameruserguide.pdf)  
2008. szeptember 15.
- [10] VIRÁGVÖLGYI PÉTER, *A tipográfia mestersége számítógéppel*, Osiris, Budapest, 2004. 2008. augusztus 15.
- [11] WETTL FERENC–MAYER GYULA–SZABÓ PÉTER,  *$\text{\LaTeX}$  kézikönyv*, Panem, Budapest, 2004.
- [12] ZSIROS ATTILA, *Dokumentumszerkesztés OpenOffice.org Writer használatával*,  
<http://www.mcl.hu/~zsati/foliak.pdf>,  
2008. augusztus 11.
- [13] *Matematikai Prezentációkészítés*  
[http://wiki.math.bme.hu/index.php/Matematikai\\_  
Prezentációkészítés](http://wiki.math.bme.hu/index.php/Matematikai_Prezentációkészítés)  
2008. szeptember 15.
- [14] [http://www.artofproblemsolving.com/LaTeX/AoPS\\_L\\_TeXer.php](http://www.artofproblemsolving.com/LaTeX/AoPS_L_TeXer.php)  
2008. szeptember 16.
- [15] <http://www.roik.bmf.hu/harp/latex/latex2html.html>  
2008. szeptember 2.
- [16] [http://www.okm.gov.hu/letolt/kozokt/kerettanterv/  
korrekturas/gimnazium/g07informatika.rtf](http://www.okm.gov.hu/letolt/kozokt/kerettanterv/korrekturas/gimnazium/g07informatika.rtf)  
2008. május 16.
- [17] [http://www.oh.gov.hu/letolt/okev/doc/erettsegi-40-2002/  
informatika-vizsgakövetelmenyek.pdf](http://www.oh.gov.hu/letolt/okev/doc/erettsegi-40-2002/informatika-vizsgakövetelmenyek.pdf)  
2008. május 16.

- [18] [http://www.sese.hu/typoblog/typeworkshop/typeworkshop\\_1.html](http://www.sese.hu/typoblog/typeworkshop/typeworkshop_1.html)  
2008. szeptember 2.