

Organizační úvod

Bude odevzdávací systém, zatím odevzdávat domácí úkoly. (Musíme splnit něco jako 7 domácích úkolů).

1 Úvod

.tex + .tfm (tex font metric = rozměry písmen) \leftarrow TEX \leftarrow DVI (formát nezávislý na OS) \leftarrow DVI moduly = DVI drivery (dvips, xdvi, pdftex (ten navíc potřebuje .tfm a fonty)).

TeX umí primitiva. V tom se ale sázení řídí špatně, tedy existují nadstavby (plainTeX, nadstavba od autora, nad nim jsou postavené conTeXt, opmac a L^ATeX) a nadstavby mimo, jako AMSTeX, XeTeX, BibTeX.

tex nebo pdftex spouští plainTeX nebo s přepínačem -ini in_iTeX, který umí zkompilevat makra do formátu.

Literatura:

- Knuth: The TeXbook: nejdřív je to takový tutoriál, potom TeX uvnitř
- Olšák: TeXbook naruby: v opačném pořadí, česky
- Knuth: TeXthe program (popsaný zdroják TeXu)

1.1 Sazba odstavce

Vstup = horizontální seznam \rightarrow (proces předělání se spouští příkazem par) zalámaný odstavec

Horizontální seznam obsahuje:

- box (h, d, w): písmenko, slitek (ligatura), hbox / vbox
- linka (h, d, w): hrule, vrule (liší se tím, kde se mohou vyskytnout, tedy hrule nepatří do horizontálního seznamu)
- discretionary break (pevné šířky): nobreak (stav bez rozdělení) pre-break (na konci řádku) post-break (na začátku dalšího), nesmí obsahovat pružné věci
- whatsit (?): přepínač jazyka, ...
- vertikální materiál (nemá vliv na horizontální sazbu, při zlomu vypadne do vertikálního seznamu), objeví se například po řádku s `\vadjust{...}`

- lepidlo (= glue) (= pružné výplňky) – věc s fixní šířkou, roztažitelností a stlačitelností (ty se udávají buď přímo v jednotkách jako px apod, nebo v jednom ze 3 nekonečen (také jednotka) fil, fill, filll).
- kern (= pevné výplňky) – věc pouze s fixní šířkou, automaticky vzniká například pro oddálení kulatých písmen / přiblížení plochých...
- penalty (= trest) – číslo, které říká, jak moc chceme / nechceme zlomit
- math on/off – zapíná a vypíná matematiku

Prvních 5 je non-discardable (jsou vidět). Ostatní discardable.

Možná místa zlomu jsou:

- glue (když před sebou má něco non-disc. a není uvnitř math)
- před kernem (když za ním je glue a není uvnitř math)
- math off (následovaný glue)
- v penaltě (ne větší než 10000)
- v discretionary breaku (ten se ale přidává až při druhém průchodu = když se nezalomí bez něho)

V zlomu se z předchozích věcí stane box řádku, u kterého se spočítají rozměry a spočítá se, jak moc je tento zlom špatný (badness), tím se vybere ten nejlepší zlom a vysází se „jeho“ řádek. Následně se zahodí všechny discardable věci a pokračuje se na dalším řádku.

Badness boxu se počítá (pokud součet nekonečných roztažitelností není nula, pak je badness 0):

$$\left\lceil 100 \cdot \left(\frac{\text{deformace}}{\text{pružnost}(\text{součet roztažitelností} / \text{stlačitelností})} \right)^3 \right\rceil$$

Roztahuje (stlačuje) se v nejvyšším nenulovém nekonečnu a to v poměru hodnot roztažnosti.

Nestlačuje se o více jak 100% (dá badness ∞).

`\hbadness` říká badness, při které už se vypisuje, že se \TeX pokusí (už ho dělá, nevybírá) vysázet box s touto badness.

`\tolerance` (při prvním průchodu `\pretolerance`, která je zpravidla o dost nižší) říká, že větší badness už se zahazuje. Při třetím průchodu už se používá, že roztažnost zvýšíme o `\emergencystretch`

Lámání odstavce funguje tak, že se prohledají zlomy (každý bod zlomu se dá považovat za vrchol grafu, hrany jsou ohodnoceny tzv. demerits (počítá se z badness a dalších věcí)).

Demerits:

$$(l + b)^2 \pm p^2 + \textit{extras}$$

(`l = \linepenalty` – přidává se vždy (aby se nelámalo všude, kde lze), `b = badness`, `p = penalt`a na níž se láme resp. `\hyphenpenalty` nebo `\exhyphenpenalty`, `extras = \adjdemerits` (10000, přidává se, když typ řádku není sousední – aby nebyl velmi roztažen, když předchozí byl stlačen) + `\doublehyphendemerits` (10000, přidává se, když vyjdou 2 neprázdné pre-breaky za sebou) + `\finaldemerits` (5000, neláme se na předposledním řádku odstavce))

Typ řádku: 0 (roztažení o více jak 100%), 1 (roztažení o 50-100%), 2 (změna max. o 50%), 3 (stlačení o více jak 50%)

Příklad (Domácí úkol)

Zkuste vymyslet, co naskládat do horizontálního seznamu, aby se mezera dala zlomit, ale zůstala na dalším řádku.

Příště tvar odstavce (`left / right`)skip, jak vzniká hor. seznam, algoritmus na pakování boxů.

Poznámka (Vzniky horizontálních věcí)

Boxy (znaky, ligatury, `\hbox to 10cm {...}`, `\vbox spread 5mm{...}` (podle v/h se skládá obsah uvnitř boxu, to nastavuje velikost, `spread` zvětšuje velikost))

Linky (`\vrule width 3pt height 10pt depth 1pt`)

Kerny (`\kern 30pt`)

Glue (`\hskip 10pt plus 2pt minus 1fil, hfil, hfill, hss`)

Penalty (`penalty 100, \nobreak, break, \allowbreak`)

Discr. break (`\discretionary{pre}{post}{no}`)

Etc.

Definice 1.1 (Box packing)

Spočítáme výsledné rozměry a deformace, pokládám věci na baseline, pomocí výšky a hloubky (příkazy `\raise 10pt \hbox{...}` a `\lower` se dají posunovat boxy vůči baseline).

Deformujeme glue.

Určí se neurčité rozměry linek.

Vertikálně se naopak pokládají referenčními body na jednu linku vlevo (zase existuje `\moveleft` a `\moveright`). Výška se pak určuje jako součet výšek a hloubek boxů uvnitř. Hloubka se počítá podle posledního boxu a pak se minimalizuje na `\boxmaxdepth` u explicitních `\maxdepth` u stránkových zlomů.

Poznámka

Na určování rozměrů se hodí tzv. podpěry (linky nulové šířky s nenulovou výškou nebo hloubkou).

Poznámka

Lze získat rozměry před i po deformaci.

Definice 1.2 (Sázení písmenek)

Font dimen: sklon std.mezera, roztažnost, smrštiteľnost, ex, em, extra mezera...

Spacefaktor je na počátku 1000. Pokud sf znaku $\neq 0$: přenastavíme (leda že by sf znaku > 1000 a my jsme < 1000 , pak nastavíme na 1000). Mezera: velikost: std.mezera + extramezera (pokud $sf \geq 1000$), roztažnost: $fd3 \cdot sf/1000$, smrštiteľnost: $fd4 \cdot 1000/sf$.

Poznámka (Nastavení pro angličtinu)

A-Z: 999, a-z: 1000, .!?: 3000, ,: 1250, (): 0

Existuje i explicitní `\spacefactor 1234`, `\frenchspacing` (nastavuje češtinu), `\nonfrenchspacing` (nastavuje angličtinu), `\spaceskip=5mm` (spaceskip překřičí font dimen aktuálního fontu) a `\xspace=3mm` (použije se, když je moc velký sf).

Poznámka (Některé příkazy)

`\line{...}` = `\hbox to \hsize{...}`

`\centerline{...}` = `\line{\hss...\hss}`

`\rlap{...}` = `\hbox to 0pt{...\hss}` (Box nulové šířky s vyčuhujícím materiálem doleva)

`\llap{...}` = `\hbox to 0pt{\hss...}` (Box nulové šířky s vyčuhujícím materiálem doprava)

Definice 1.3 (Dělení slov)

Pro každý jazyk má \TeX trie, jak slovo dělit.

Existuje makro `\chyp`, které přepne do češtiny.

Definice 1.4 (Módy fungování v \TeX u)

(Přesněji řečeno módy hlavního procesoru)

- Vertikální hlavní (stránkový)
- Vertikální vnitřní (`\vbox`)
- Horizontální odstavcový (zalamování)
- Horizontální vnitřní / restricted (`\vbox`)

- Matematický vnitřní (\$)
- Matematický display (\$\$)

Na začátku je \TeX v hlavním vertikálním módu. Teprve ve chvíli, kdy najde něco, co by mělo být v odstavci (písmenko, `\noindent`, `\indent`, `\leavevmode` (jako písmenko), `\hskip`, `\vrule`), tak se \TeX přesune do odstavcového horizontálního.

Zpět se přesouvá příkazy (`\par` (tj. i 2 odřádkování), vertikálními povely: `\par`, `\vskip`, ..., } ukončující `\vbox`), což vyvolá odstavcový zlom a vrácení se do hlavního vertikálního.

Obdobně ostatní přechody (pozor, lze přecházet i z Vertikálního vnitřního do odstavcového, naopak nelze přecházet z vertikálních do matematických, tam se automaticky přechází přes horizontální).

Poznámka (Co dostane lámací algoritmus)

Na začátku prázdný *box* šířky `\parindent`. Následuje horizontální materiál odstavce a „ocásek“, ve kterém je `\unskip` (odstranění poslední mezery), `\nobreak`, glue velikosti `\parfillskip = 0 pt` a `\break`.

┌
Poznámka (Co lze)

Přenastavit `\parfillskip = \parindent` (pak bude odstavec, když to vyjde, symetrické).

Přenastavit `\parfillskip = 1cm plus 1fil` (např. když máme malou mezeru mezi odstavci a chceme uživatele upozornit na konec odstavce, i když vychází do konce řádku).

Poznámka (Sestavení řádku)

Horizontální materiál řádku se obalí `\leftskip = 0pt` zleva a `\rightskip = 0pt` zprava a zavře se do `\hbox` velikosti `\hsize`.

Sázení na praporek lze vytvořit tím, že nastavíme `\rightskip = 0pt plus 1 fil`, ale pak se budou řádky snadno lámat (nebudou se rozdělovat slova, budou kratší řádky). Správně na to existuje makro `\raggedright`, které udělá `\rightskip 0pt plus 4em\spaceskip=... \xspacesk` (nastaví mezislovní a písmenné mezery na pevné, aby se neroztahovali podle smršťování a roztahování té mezery na konci).

Centrování `\leftskip = \rightskip = 0,4 plus 2em\parfillskip = 0pt`.

Poznámka (Tvar odstavce)

Vykousnutí se nastavuje `\handindent` (rozměr, o kolik se odsadí) a `\hadgafter` (číslo, kolik řádků se odsadí), když se nastaví záporné hodnoty, vykusuje se intuitivně ostatní rohy odstavce. Na konci odstavce se nuluje.

Následuje `\parshape = n p1 w1 ... pn wn` (kolik se má odsadit, o kolik které, poslední se opakuje do nekonečna). Také se nuluje.

Když zrovna nejsme ve vertikálním módu, tak se v `\prevgraf` uchovává počet řádků v předchozím odstavci.

Existuje makro `\everypar`, které spustí nastavený kód každý odstavec.

Můžeme si objednat zmenšení / zvětšení počtu řádků `\looseness = n` (- je kratší, pokud nelze vyplnit, bude ignorováno).

Poznámka (Výsledek lámání odstavce: vertikální materiál) • \forall řádek jako box + posunutí referenčního bodu (žádné glue).

- Dále z boxů vypadají vertikální věci (`\vadjust`, `mark`).
- Penalty mezi řádky (`\interlinepenalty = 0 + clubpenalty = 150` (po prvním řádku) + `widowpenalty = 150` (před posledním řádkem) + `\brokenpenalty = 100` (po pre-break) + `\displaywidowpenalty` (aby nebyla osamocená display matematika)).
- Ještě se objeví vertikální (zde řádkové) mezery, ale ty probereme zvlášť.

Definice 1.5 (Řádkování)

Algoritmus, aby se pokud možno dodrželo řádkování (ale řádky mohou být různě široké). Řídí se 3 parametry: `\baselineskip` (glue), `lineskiplimit = 0pt` (dimen) a `lineskip = 1pt` (glue).

Vypočítá mezeru jako $skip = bls - d_{horni} - h_{spodni}$. Pokud vyjde $skip < lsl$, nastaví se $skip = ls$. (Při více stránkách není dobré nastavovat pružnost těchto mezer).

`vskip`, `kern`, `penalty` ignorujeme, hrule algoritmus potlačí.

┌ *Poznámka* (Jak je to doopravdy)

V registru `\prevdepth =` hloubka posledního boxu (`-1000pt`: algoritmus potlačen), linka nastaví právě ten dolní limit.

`\nointerlineskip` je `\prevdepth = -1000pt`. `\offinterlineskip` úplně zastaví tento algoritmus `\baselineskip = -1000pt`, `\lineskip = 0`, `\lineskiplimit = \maxdimen`

Poznámka (Usazení 1. řádku na stránce (pouze hlavní vertikální mód))

Snažíme se spočítat glue tak, aby výška mezery + výška 1. řádku vyšla `\topskip`, ale není nikdy záporný.

Rozdíl proti řádkovému: nemáme limit (vždy je 0pt) a uvažujeme linky.

Poznámka (Ještě k předchozímu)
Na začátku odstavce se vloží `\parskip`.

2 registry

Definice 2.1

Registry jsou zabudované (konkrétní počet; pojmenované; spousta nastavení, o kterých jsme mluvili) a uživatelské (0...255 každého typu, často (u dalších „TeXů“) i více).

Typy:

- `\count` – číslo (31 bitů + znaménko)
- `\dimen` – rozměr (30 bitů + znaménko ve $\text{sp} = 2^{-16} \text{pt}^a$, tj. 14 celá část, 16 desetinná)
- `\skip` – roztažnost (13+16 bitů)
- `\muskip` – matematický (speciální jednotky)
- `toks`, `box`, ...

Registry se obnovují po konci grupy na začínající stav.

Použití: lze do nich dosazovat (`\count74=32b`, `\parskip=10pt` (rovnítka lze vyměnit za mezeru, či vynechat)), lze ho použít jako jednotky, vypsát ho (`\the\count5`) (do pdf), vypsát ho (`\showthe\count5`) (do logu), použít jako pointer (`\count\count5`), automaticky konvertovat `dimen` \rightarrow `skip` nebo `skip` \rightarrow `dimen` \rightarrow `count`.

^a $\text{pt} = 1\text{in}/72,27$

^bČíslo lze napsat číslicemi s desetinnou tečkou, apostrof a osmičková soustava, 2 apostrofy a šestnáctková ve velkých písmenech, obrácený apostrof znak resp lomítka znak, hodnota registru a backslash pojmenovaný znak (pomocí `*chardef*ch=kód` (`*`=lomítka) to však TeX užívá spíše uvnitř).

Definice 2.2 (Aritmetika)

`\advanced registr by hodnota` (by lze vynechat nebo napsat BY)
`\multiply` (pouze celými čísly)
`\divide` (pouze celými čísly), zaokrouhluje se k nule

Definice 2.3 (Alokace registrů)

- `count 0...9` = číslo stránky
- `box 255` = přenos obsahu do output rutiny

- `reg. 0...9` = pracovní (krom čísel stránek)
- `\countdef\jmeno=cislo` – nastavuje přezdívku za registr s číslem `cislo`
- `\newcount\pocitadlo` – (plain) alokuje nějaký registr (interně `\countdef\pocitadlo`)
- `\newinsert\...` – (plain) alokuje vše, co potřebuje na insert, viz dále

Definice 2.4 (Boxový registr)

Obsahuje nic, hbox nebo vbox.

Lze ho nastavit (`\setbox0=\hbox{...}`), přemístit na aktuální místo (`\box0`), vložit na aktuální místo (`\copy0`), přemístit / vložit jejich obsah na aktuální místo (`\unhbox0`, `unvbox`, `unhcopy`, `\unvcopy`), měřit / měnit rozměry (`\wd0`, `\ht0`, `\dp0`)^a, (`\showbox0`), (`\newbox\cs`).

^atoho využívají plainová makra `*phantom{...}`, `*hphantom{...}`, `*vphantom{...}` (*=lomítko), které vytvoří prázdné boxy velikosti jejich obsahů.

3 Stránkový zlom

Definice 3.1 (Obsah vertikálního seznamu)

- `box`
- `linka`
- odkaz na insert (plovoucí obsah)
- `mark`
- `whatsit` (třeba zápis do souboru `\write`, `\special` viz dále)
- `glue`
- `kern`
- `penalty`

Prvních 5 je non-discardable.

Definice 3.2 (Stránkový zlom)

Místa zlomu: `glue`, před nímž je non-disc., `kern` za `glue`, `penalty` < 10000 .

Nebyl dostatek paměti na obtížnější, tedy se postupně přidávají prvky, počítá se `cost` ta

je na začátku 100000, protože by se obsah moc roztáhl, pak jsou rozumné a někdy dojde na nekonečno, kde algoritmus najde zpětně nejlepší zlom (pamatuje si ho, ze stejných vybere ten poslední = nejplnější) a tam zlomí.

Cost^a: 1) $b < \infty, p \leq -10000, q < 1000 : c := p$, 2) $b < 10000, p \in (-10000, 10000), q < 10000 : c := b + p + q$, 3) $b = 10000$ (*underfull*), ostatní konečné jako v 2) : $c := 100000$, 4) jinak: $c := +\infty$.

Pamatuje si `\pagetotal`, kde si pamatuje, co už má na stránce, `\pagestretch`..., kde si pamatuje počty roztažností, a `\pagegoal`, kde si pamatuje výšku (bez plovoucích tedy `\vsize`).

^a $c = \text{cost}$, $b = \text{badness}$, $p = \text{penalta}$, $q = \text{dodatečná penalta}$

Příklad

Plain má makro `\raggedbottom`, který nechá vlát dole (pružný konec stránek), vytvořte ho.