

\LaTeX & TABULKY & TIPY & TRIKY \\\

JIŘÍ RYBIČKA (CZ)

Abstrakt. Sazba tabulek představuje zejména v odborných textech velmi frekventovanou součást. Základní přístup ke tvorbě tabulek se významně liší, použijeme-li majoritně používaný systém pro zpracování textů (např. Open Office Writer) oproti systémům pracujícím na principu \TeX . Většina uživatelů pohlíží na tabulku jako na grafický objekt, proto jim plně vyhovuje interaktivní přístup, jehož základní nevýhodou je nízká nebo žádná automatizovatelnost a problematická typografická kvalita. Dávkový přístup implementovaný v systému \LaTeX naproti tomu komplikuje vytvoření požadovaného tvaru tabulky. Článek se zabývá některými možnostmi usnadnění tvorby tabulek při zachování nebo rozšíření automatizovatelnosti a současného dosažení požadované typografické kvality výsledku.

Klíčová slova. Tabulka v \LaTeX u, `tabular`, `tabbing`, `parbox`, `catcode`.

\LaTeX & TABLES & TIPS & TRICKS \\\

Abstract. Typesetting of tables is frequently used component of texts especially scientific articles and books. The basic approach to creating tables differ significantly if we use majority systems for text processing (eg. Open Office Writer) and systems operating on the principle of \TeX . Most users considers the table as a graphical object, so they fully complies with interactive approach whose main drawback is low or no automatizing and problematic typographic quality. Batch approach implemented in a system \LaTeX on the other hand is complicated to create the desired shape of the table. Article deals with some tips to facilitate the creation of tables while achieving the automatical processing and the desired typographical quality of the result.

Keywords. \LaTeX table, `tabular`, `tabbing`, `parbox`, `catcode`.

1. Požadavky na sazbu tabulek

Sazba tabulek představuje zejména v odborných textech zcela nezbytnou součást konstrukce dokumentu. Stejně jako u jiných prvků jsou i na tabulky kladeny určité požadavky.

Tyto požadavky můžeme rozdělit na typografické (to by mělo být prvotní hledisko) a technické. Dostatečně podrobně tuto problematiku rozebírá ve své diplomové práci Talandová (2006). Na stranách 17–22 uvádí přehled typografických zásad a pravidel, požadavky na technické zpracování shrnuje na stranách 23–32. Základním výsledkem těchto dvou částí je konstatování, že typografické systémy tabulky samozřejmě podporují, ale nejsou obvykle vedeny typografickými zásadami; rovněž přístupy k řešení tabulek jsou velmi rozdílné a dosažení téhož efektu si může vyžádat diametrálně odlišnou pracnost.

Správně typograficky upravená tabulka je především přehledná, vyvážená, má správně zarovnané údaje a čtenáři přináší maximální užitek. Dosažení tohoto cíle vyžaduje obvykle kombinaci mnoha prvků a určitou volnost při jejich využívání.

Od doby kovové sazby se výrazně rozšířily možnosti sazby tabulek. S rozvojem počítačové grafiky se doména tabulkových nástrojů postupně přesunula do této oblasti, čímž se zcela uvolnily možnosti využívání dříve zcela nedostupných prvků – linek nejrůznějších typů, tlouštěk a barev, používání podbarvených polí, seskupování tabulkových polí do libovolných skupin, manipulace s interním prostorem a materiálem v polích atd.

Samozřejmě že ne všechny tyto nástroje jsou využívány k dosažení typograficky optimálního výsledku. To však není obvykle vnímáno jako nedostatek, zatímco určitá komplikovanost nástrojů v některých systémech bývá značnou překážkou pro uživatele.

1.1. Malovat, nebo programovat?

Základním přístupem, který je uživateli silně preferován, je interaktivní (grafický) návrh tabulky – malování, zahrnující téměř neomezené možnosti volby způsobu ohraničení a podbarvení, manipulace se šířkami a výškami polí, vizuální značkování obsahu polí a snadné přecházení mezi jednotlivými poli. Je pochopitelné, že jednoduchost tohoto přístupu, neustálá vizuální kontrola výsledku a přítomnost řady nástrojů příliš nedávají šanci systémům postaveným na principu $\text{T}_{\text{E}}\text{X}$ – tedy programování, jejichž přístup vyžaduje neinteraktivní práci, vkládání příkazů a relativně komplikované řešení situací, které v interaktivní verzi nepřinášejí prakticky žádné problémy.

Na obhajobu neinteraktivních systémů lze však říct, že zdaleka ne vždycky je jejich použití nevýhodné. Naopak, existuje řada případů, kdy vhodným využitím bohatého aparátu tvorby maker dosáhneme vytčeného cíle rychleji, s minimálním úsilím při případných úpravách vzhledu a se zajímavými možnostmi integrace tabulek do okolního materiálu dokumentu. Nezanedbatelná není ani typografická kvalita, v některých situacích má dokonce neinteraktivní systém nástroje, které nemají v interaktivních protějšcích žádnou obdobu.

Základní manipulace s předdefinovanými příkazy lze zjistit z mnoha zdrojů a učebnic, jednou z možností je učebnice Rybičky (2003). V těchto textech jsou však uvedeny do značné míry jen základní rekvizity, jejichž použití v mnoha běžných dokumentech s sebou nese největší nevýhodu neinteraktivních systémů – těžkopádný a nepřehledný zápis.

Podívejme se tedy na několik drobných tipů, které nám mohou pomoci v neinteraktivním přístupu tvorby tabulek v systému $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ a jeho následnících. Tipy vycházejí z dlouholetých a osvědčených zkušeností s dokumenty různých druhů.

2. Některé možnosti řešení

Nerad používám cizí makra.

Petr Olšák

Nerad znovu vynalézám kolo.

Petr Sojka

Uvedené citáty možná nejsou zcela autentické, ale vyjadřují dvojí diametrálně odlišný přístup k řešení problémů – buď si všechno vyrobím sám, umím to velmi dobře a mám nad celým řešením plnou kontrolu, nebo využiji již existujících balíčků, které ovšem mohou mít netušené vnitřní vazby a skryté zrady a často je budu muset přizpůsobit konkrétní situaci.

Praktické případy však pravděpodobně leží někde mezi těmito krajními mezemi. Je vhodné vědět, že podobný problém už někdo řešil (a možná dobře vyřešil), na druhé straně místo dlouhého hledání vhodného balíčku je daleko efektivnější napsat dvě nebo tři vhodná makra a danou konkrétní situaci tím vypořádat. Nebude to třeba řešení obecné, ale o to mnohdy vůbec ani nejde.

V každém případě přístup neinteraktivní, tedy „programátorský“, má svou největší výhodu v automatizovatelnosti. Vhodnou cestou je prakticky vždy vytvoření nějakých vlastních příkazů, které zefektivní zápis zdrojového textu a umožní také často velmi výrazně usnadnit jakékoliv další změny a zásahy. Nebraňme se tedy vlastním příkazům – v opačném případě je vhodnější zůstat u malování (a také asi u jiného systému).

2.1. Obejdeme se bez tabular?

Když se řekne „tabulka v L^AT_EXu“, prakticky každému se ihned vybaví nejfrekventovanější nástroj – prostředí `tabular`. Je však vždy nutné tento do značné míry komplikovaný přístup použít?

Pro sazbu tzv. otevřených tabulek (v nich stačí obvykle pouze dvě vodorovné linky, jinak nevyžadují žádné orámování) můžeme použít prostředí `tabbing`, v jednoduchých případech vystačíme pouze s boxy, u nichž stanovíme vhodnou šířku. Příklad – pár údajů o teplotách naměřených dnes v 7 hodin ráno. Zdrojový text může mít tento tvar:

```
\def\radek#1;#2.{\noindent\makebox[50mm][l]{#1}% 1. sloupec
\makebox[10mm][r]{#2}\par} % 2. sloupec

{\bfseries \radek Město; Teplota [^\circ$C].}
\smallskip\hrule width 60mm\smallskip
\radek Ostrava; 12,8.
\radek Olomouc; 14,2.
\radek Bruntál; 9,8.
```

```

\radek Brno; 13,3.
\radek Trenčín; 15,5.
\radek Žilina; 14,3.
\radek Banská Bystrica; 11,6.
\smallskip\hrule width 60mm

```

Po vysázení dostaneme:

Město	Teplota [°C]
Ostrava	12,8
Olomouc	14,2
Bruntál	9,8
Brno	13,3
Trenčín	15,5
Žilina	14,3
Banská Bystrica	11,6

2.2. Zalamování řádků v buňkách

Zcela přirozeným požadavkem je, aby se materiál v buňce tabulky mohl lámat na řádky. To ovšem ani v prostředí `tabbing`, ani v prostředí `tabular` běžně možné není. Nepátřejme nad příčinami tohoto na první pohled málo pochopitelného stavu a pojďme se podívat, jak tuto nevýhodu obejít. Budeme se zabývat prostředím `tabular`.

Obsahem každého tabulkového pole je tzv. LR-box, tedy box v omezeném horizontálním režimu. Jeho součástí je nejčastěji jednořádkový text. Potřebujeme-li více řádků, musíme si pomoci nějakým trikem.

Nejjednodušší možností je použít uvnitř pole další prostředí `tabular`, uvnitř něho pak můžeme vkládat více řádků. Toto vnořené pole však musíme poněkud ošetřit, aby výsledek splňoval alespoň základní vzhledové požadavky.

Vložená tabulka je jednosloupcová, ale potřebujeme u ní odstranit mezisloupcovou mezeru na levé i pravé straně, aby se tato mezera nescítala s mezisloupcovou mezerou pole, do něhož tabulku vkládáme. Druhá úprava spočívá v úpravě řádkování vnořené tabulky – dost často potřebujeme, aby řádky byly o něco hustší než řádky okolní tabulky, protože implicitní světlo v tabulkových polích je příliš malé a potřebujeme je roztáhnout. To ovšem neplatí pro vnořené tabulky.

```

% víceřádkové tabulkové pole:
\def\pole#1#2{{\def\arraystretch{1}%
  \begin{tabular}{@{}#1@{}}#2\end{tabular}}}

% tabulkové pole se zarovnáním na horní okraj:
\def\polet#1#2{{\def\arraystretch{1}%
  \begin{tabular}[t]{@{}#1@{}}#2\end{tabular}}}

```

```
\def\arraystretch{1.3}% běžné světlo v tabulkách
```

```
% příklad tabulky:
```

```
\begin{tabular}{|l|c|} \hline
\bfseries Město & \bfseries\pole c{Teplota\\{}}[$^\circ$C]}
\\ \hline \hline
Ostrava & 10,8 \\ \hline
Olomouc & 9,2 \\ \hline
Žilina & 11,3 \\ \hline
\pole l{Banská\\ Bystrica} & 11,6 \\ \hline
\end{tabular}
```

Po vysázení dostaneme (v první tabulce je makro `\pole`, ve druhé je použito makro `\polet`):

Město	Teplota [°C]
Ostrava	10,8
Olomouc	9,2
Žilina	11,3
Banská Bystrica	11,6

Město	Teplota [°C]
Ostrava	10,8
Olomouc	9,2
Žilina	11,3
Banská Bystrica	11,6

Všimněte si, jak je v těchto tabulkách rozdílný prostor nad vnořeným polem. V prvním případě je mezera menší (neuplatní se zde nastavení `\arraystretch`), zatímco ve druhém případě se vlivem zarovnání na horní řádek víceřádkového pole uplatní nastavení řádková mezera.

Nepřípustné zarovnání čísel ve druhém sloupci vyřešíme v následující části.

2.3. Jak na číselná data

Požadavek zarovnání čísel tak, aby stejné řády byly pod sebou, je zcela nesporný. Máme-li prezentovat data v tabulce, děláme to právě proto, aby čtenář **na první pohled** viděl, jaké jsou mezi číselnými hodnotami rozdíly.

Podíváme-li se kolem sebe na nejrůznější dokumenty pocházející z nejrůznějších systémů, můžeme konstatovat, že se v tomto směru objevuje masivní množství chyb. Jedna z nejčastějších je demonstrována na předchozím příkladu: sloupec je zarovnán na střed, ale čísla nemají stejné šířky, takže různé řády plavou různě

pod sebou. Tím se přehled o velikostech jednotlivých hodnot stírá, čímž je vlastně popírán hlavní důvod, proč se data do tabulek umisťují.

Řešením jistě je sloupec upravit na pravý okraj. Vznikne tím tvar, který jsme uvedli v prvním příkladu jednoduché tabulky: Popisek sloupce je zpravidla podstatně delší než číselné údaje, vzniká tím tedy esteticky nepřipustný zející volný prostor s daty namačkanými na pravé čáře. Správně by data měla splňovat dvě podmínky současně: měla by být na střed, ale stejnými řády pod sebou.

Toho lze docílit mimo jiné tím, že čísla budou mít stejný počet číslic a všech ostatních prvků (desetinných čárek, znamének apod.). Prvky, které absentují, tedy doplníme vhodnými mezerami. Postupovat však nebudeme tak, jak to dělají zcela laičtí uživatelé v interaktivních systémech, že budeme mlátit do mezerníku tak dlouho, až budeme na obrazovce zhruba spokojeni a pak se raději ani nebudeme dívat na tištěný výstup. Mezery dodáme ve zcela přesných velikostech.

V některých systémech se za tímto účelem vyskytuje tzv. figurální mezera – tedy mezera o šířce číslice. Číslice, které lze použít v tabulkách, mají totiž identické šířky – jsou to tzv. neproporcionální číslice. Kde v L^AT_EXu máme figurální mezera? Lze jí docílit například příkazem `\hphantom{0}`.

Ano, nic složitého, ale psát do předchozí tabulky u „kratšího“ čísla (v případě složitějších tabulek do každého řádku) text `\hphantom{0}9,2` je přinejmenším těžkopádné. Už i tak málo přehledný zápis tabulky se tímto „instrumentem“ pěkně zamíchá. Vhodným řešením je využít k tomuto účelu jednoznakový příkaz (aktivní znak), který umožní i snadnou optickou kontrolu, pokud si zápis tabulky vhodně zformátujeme v editoru. Definujme tedy figurální mezera jako například znak vykřičník, který předtím učiníme aktivním znakem:

```
\catcode'\!=\active
\def!{\hphantom{0}}
```

Tabulku upravíme tak, aby ve zdrojovém textu bylo snadno vizuálně kontrolovatelné, že všechna čísla jsou stejně široká:

```
\begin{tabular}{|l|c|c|} \hline
\bfseries Příčina ukončení testu &
\bfseries \pole c{Absolutní \ \ četnost} &
\bfseries \pole c{Relativní\ \ četnost [%]} & \\\hline
Opisování z taháku & 180 & !25,0 & \\\hline
Použití mobilního telefonu & !45 & !6,3 & \\\hline
Vyžádání pomoci od spolužáka & 405 & !56,2 & \\\hline
Vložení řešení do kalkulačky & !90 & !12,5 & \\\hline
\bfseries Celkem $N$ & 720 & 100,0 & \\\hline
\end{tabular}
```

Po vysázení dostaneme:

Příčina ukončení testu	Absolutní četnost	Relativní četnost [%]
Opisování z taháku	180	25,0
Použití mobilního telefonu	45	6,3
Vyžádání pomoci od spolužáka	405	56,2
Vložení řešení do kalkulačky	90	12,5
Celkem N	720	100,0

Mírně odlišný problém je použití znaménka minus, jehož výskyt nesmí narušit zarovnání na stejné řády. K tomu účelu si můžeme vytvořit jednoduché makro pro box s nulovou šířkou:

```
\def\mm{\makebox[0pt][r]{$-$}}
```

a použít je v následujícím příkladu:

```
\begin{tabular}{|l|c|} \hline
\bfseries Hodnocené příčiny & & \\
\bfseries \pole c{Relativní\četnost [%]} & & \\
Vliv speciálního předmětu & 25,0 & \\
Negativní zkušenosti ze SŠ & !\mm6,3 & \\
Penalizace u hodnocení prací & 56,2 & \\
Tendence usnadnit si práci & \mm12,5 & \\
\bfseries Celkový výsledek & 62,4 & \\
\end{tabular}
```

Po vysázení dostáváme:

Hodnocené příčiny	Relativní četnost [%]
Vliv speciálního předmětu	25,0
Negativní zkušenosti ze SŠ	−6,3
Penalizace u hodnocení prací	56,2
Tendence usnadnit si práci	−12,5
Celkový výsledek	62,4

2.4. Poznámky v tabulce

Navážeme na předchozí rekvizity podobným řešením dalších přidavných symbolů, které se vyskytují u prezentovaných dat. Jedná se nejčastěji o různé poznámkové

symboly – hvězdičky, mečíky, písmena. Samotná poznámka by pak měla být sázena jako součást tabulky a měla by mít označení, které se nemíchá s běžnými poznámkami pod čarou. Často se odkazy na tyto lokální poznámky řeší nečíslně, aby nemohlo dojít k záměně s údaji v tabulce.

Základní prvek je podobně jako v případě znaménka minus box s nulovou šířkou, tentokrát se zarovnáním doleva:

```
\def\hve{\makebox[0pt][l]{*}}
```

Analogicky definujeme podobná makra pro případné další značky.

Lokální tabulkovou poznámku lze řešit jako zvláštní řádek tabulky s odpovídajícím zarovnáním přes všechny její sloupce. K tomu si vyrobíme makro:

```
\def\tabpozn#1#2#3{\multicolumn{#1}{@{}l@{}}{\%
\parbox[t]{#2}{\footnotesize\raggedright #3}}}
```

a použijeme je v následujícím příkladu:

```
\begin{tabular}{|l|c|} \hline
\bfseries Hodnocené hypotézy & & \\
\bfseries Hodnota kritéria & & \\
H$_0$ Podvádění je naučené & 143,8\hve & \\
H$_0$ Plagiátorství lze snížit výukou & !74,4\hhve & \\
H$_0$ Penalizace je dostatečná & !!7,9\ha & \\
H$_0$ Pomůže softwarová detekce & !13,4\hb & \\
\end{tabular}
```

\$^a\$,hypotéza nebyla zamítnuta;\\$ ^b\$,hypotéza byla zamítnuta.}

Po vysázení dostaneme tvar:

Hodnocené hypotézy	Hodnota kritéria
H ₀ Podvádění je naučené	143,8*
H ₀ Plagiátorství lze snížit výukou	74,4**
H ₀ Penalizace je dostatečná	7,9 ^a
H ₀ Pomůže softwarová detekce	13,4 ^b

* pro $p < 0,05$; ** pro $p < 0,01$;

^a hypotéza nebyla zamítnuta;

^b hypotéza byla zamítnuta.

2.5. Odstavcové buňky

Dalším frekventovaným požadavkem je umístění souvislého textu do tabulkového pole. Tento stav je obvykle v interaktivních systémech implicitní. Sazba odstavcového textu do tabulkových polí má však své zvláštnosti. Obvykle se jedná o sazbu do užšího rozměru, je tedy nezbytné, aby byla zarovnána na prapor, někdy na střed. Prostředí `tabular` však nabízí jako jedinou možnost definovat sloupec se

zarovnáním do bloku. Vypomůžeme si však jednoduchým makrem, které nám v řadě případů velmi usnadní práci – tabulkovým odstavcem řešeným pomocí parboxu:

```
\def\strdn{\vrule depth .5em width 0pt}
\def\txpar#1#2{\parbox[t]{#1}{\raggedright #2\strdn}}
```

Vertikální čára nulové šířky vkládaná na konec odstavcového materiálu zde zastupuje činnost řádkování tabulky definované pomocí `\arraystretch`, protože to zde nefunguje. Lokálními redefinicemi makra `\strdn` můžeme snadno ovlivňovat situaci v jednotlivých tabulkách dokumentu. Podobně lze řešit i prostor v horní části pole. Samotné makro `\txpar` je poněkud těžkopádné. Na jeho základě si však můžeme v každé tabulce definovat vhodné následníky, jak je ilustrováno na následujícím příkladu. Makro `\tx` představuje materiál, který je ve všech čtyřech sloupcích tabulky stejně široký.

```
\def\tx#1{\txpar{0.2\textwidth}{#1}}

\begin{tabular}{|l|l|l|l|}\hline
\tx{\bfseries Humánní etologie} & & & & \\
\tx{\bfseries Výzkumy komunikace} & & & & \\
\tx{\bfseries Psychologie} & & & & \\
\tx{\bfseries Pedagogika} & & & & \\
\tx{\bfseries Trevarthen:} & & & & \\
Teorie intersubjektivit & & & & \\
Reciprocita & & & & \\
Protokonverzace & & & & \\
Vrozené motivace ke komunikaci & & & & \\
\tx{\bfseries Jakobson:} & & & & \\
Lingvistika & & & & \\
Rituály kontaktu & & & & \\
{\bfseries Watzlawick:} & & & & \\
Pragmatika komunikace & & & & \\
Videofeedback & & & & \\
\tx{\bfseries Bowlby:} & & & & \\
Teorie citové vazby & & & & \\
{\bfseries Vygotsky, Stern:} & & & & \\
Vývojové teorie & & & & \\
Atribuční teorie & & & & \\
\tx{\bfseries Tausch / Tausch, Rogers:} & & & & \\
Humanistická pedagogika & & & & \\
{\bfseries Bruner:} & & & & \\
Kulturní teorie učení & & & & \\
\end{tabular}
```

Po vysázení dostaneme tento výsledek:

Humánní etologie	Výzkumy komunikace	Psychologie	Pedagogika
Trevarthen: Teorie intersubjektivit Reciprocita Protokonverzace Vrozené motivace ke komunikaci	Jakobson: Lingvistika Rituály kontaktu Watzlawick: Pragmatika komunikace Videofeedback	Bowlby: Teorie citové vazby Vygotsky, Stern: Vývojové teorie Atribuční teorie	Tausch / Tausch, Rogers: Humanistická pedagogika Bruner: Kulturní teorie učení

Zde je „nastřelena“ šířka sloupce na hodnotu $0.2 \cdot \text{textwidth}$, ale můžeme postupovat i daleko promyšleněji. Chceme (a to je také dost častý požadavek), aby tabulka byla široká přesně jako šíře sazby. Necháme si tedy odpovídající šířku spočítat například takto:

```
\newlength{\lntxpar}
\lntxpar=\textwidth % vyjdeme ze šířky sazby
\advance\lntxpar by -8\tabcolsep % odečteme mezisloupcové mezery
\divide\lntxpar by 4 % vydělíme počtem sloupců
\def\tx#1{\txpar{\lntxpar}{#1}}
```

Stejnou tabulku pak dostaneme v následujícím provedení:

Humánní etologie	Výzkumy komunikace	Psychologie	Pedagogika
Trevarthen: Teorie intersubjektivit Reciprocita Protokonverzace Vrozené motivace ke komunikaci	Jakobson: Lingvistika Rituály kontaktu Watzlawick: Pragmatika komunikace Videofeedback	Bowlby: Teorie citové vazby Vygotsky, Stern: Vývojové teorie Atribuční teorie	Tausch / Tausch, Rogers: Humanistická pedagogika Bruner: Kulturní teorie učení

2.6. Umístění tabulky do dokumentu

Tabulka jako celek není pevnou součástí textu. Protože by měla být přehledná, je nevýhodné, pokud systém dovolí její běžný zlom na koncích stránek. Každá tabulka také musí mít popisek systematicky umísťovaný buď nad tabulku, nebo pod tabulku (v celém textu však jednotně). Tyto popisky se často přenášejí i do seznamů tabulek.

Pro umístění tabulky do dokumentu lze s výhodou využívat plovoucího prostředí `table`. V základním tvaru tohoto prostředí je k dispozici i příkaz `\caption` definující popisěk tabulky. Implicitní tvar obojího nám však nemusí vyhovovat a navíc je vhodné mít možnost všechny tabulky dokumentu centrálně ovládat (nastavit vhodné písmo, mezery apod.).

Všechny tyto důvody vedou k vytvoření vlastního makra pro vkládání tabulek. Jeho zjednodušený tvar může být například následující:

```
% fonty:
\def\tabpopisfnt{\small\sffamily\itshape}
\def\tabulfnt{\small\sffamily}
% popisěk:
\def\mujpopisek#1{\refstepcounter{table}
  \parbox{\textwidth}{\raggedright \tabpopisfnt #1}}
% samotná tabulka:
\long\def\tabulka#1#2\endtab{\begin{table}[htb] %plovoucí
  \centering % materiál na střed
  \tabulfnt % písmo pro tabulky
  #2 % umístění materiálu tabulky
  \par\medskip\mujpopisek{#1} % popisěk pod tabulkou
  \end{table}}
```

Použití makra demonstruje následující příklad, v němž využijeme řadu uvedených rekvizit:

```
\tabulka{Plánování podnikání a~hodnocení přípravy na vysoké škole}
\newsavebox{\pombox}
\lntxpar=\textwidth
\sbox{\pombox}{Celkem}
\advance\lntxpar by -\wd\pombox
\advance\lntxpar by -8\tabcolsep
\divide\lntxpar by 3
\def\tx#1{\txpar{\lntxpar}{#1}}
\begin{tabular}{|l|c|c|c|} \hline
& \tx{\bfseries Uvažujete o~tom, že byste ... podnikali?}
& \tx{\bfseries Myslíte si, že Vás... jiné i~na to, ... }
& \tx{\bfseries Máte na své fakultě ... předmětů?} \\ \hline
Ano & !3 !(7,3\,\%) & !7 !(17,1\,\%) & !38 !(92,7\,\%) \\ \hline
Ne & !27 !(65,9\,\%) & !4 !(9,8\,\%) & !10 !(0,0\,\%) \\ \hline
Nevím & !11 !(26,8\,\%) & !30 !(73,2\,\%) & !13 !(7,3\,\%) \\ \hline
Celkem & !41 (100,0\,\%) & !41 (100,0\,\%) & !41 (100,0\,\%) \\ \hline
\end{tabular}
\endtab
```

Po vysázení dostáváme tabulku č. 1, ve které je změněno písmo a popisek tabulky je umístěn pod ní.

	Uvažujete o tom, že byste v budoucnu (horizont 4 let) podnikali?	Myslíte si, že Vás fakulta dokáže připravit mimo jiné i na to, abyste v budoucnu sami začali podnikat?	Máte na své fakultě možnost studia ekonomických předmětů?
Ano	3 (7,3 %)	7 (17,1 %)	38 (92,7 %)
Ne	27 (65,9 %)	4 (9,8 %)	0 (0,0 %)
Nevím	11 (26,8 %)	30 (73,2 %)	3 (7,3 %)
Celkem	41 (100,0 %)	41 (100,0 %)	41 (100,0 %)

Tab. 1: Plánování podnikání a hodnocení přípravy na vysoké škole

3. Závěr

Praxe přináší stále nové a nové náměty, které je potřeba řešit. Podobným způsobem bychom tedy mohli pokračovat ještě dále – samostatnou kapitolou jsou barvy v tabulce (ohraničující čáry i tabulková pole), řadu problémů je potřeba řešit, když se materiál tabulky nevejde do stanoveného prostoru atd.

Hlavním cílem článku však není předkládat množství hotových řešení – důležitý je zejména postup: pokud možno jednoduchými prostředky dosáhnout požadovaného efektu, byť se jedná o zcela lokální řešení, jehož obecná podoba v nějakém balíčku by byla nesrovnatelně náročnější a koneckonců i pro uživatele těžkopádnější. Ukazuje se, že cesta, kterou hodně prosazuje P. Olšák například svým projektem OPMac (Olšák, 2014), je schůdná i pro méně zdatné uživatele $\text{T}_{\text{E}}\text{X}$ u a jemu příbuzných systémů. Investice do naučení tvorby maker se podle praktických zkušeností určitě bohatě vyplatí.

Reference

- [1] OLŠÁK, P.: *OPmac – rozšiřující makra plain $\text{T}_{\text{E}}\text{X}$ u*, 2014, <https://math.feld.cvut.cz/ftp/olsak/opmac/opmac-u.pdf>.
- [2] RYBIČKA, J.: *L $\text{T}_{\text{E}}\text{X}$ pro začátečníky*, Brno, KONVOJ 2003, ISBN 80-7302-049-1.
- [3] TALANDOVÁ, P.: *Přístupy ve zpracování tabulek v systémech DTP*, diplomová práce, MZLU v Brně, 2006.

Kontaktní adresa

doc. Ing. Jiří Rybička, Dr., Ústav informatiky, Provozně ekonomická fakulta, Mendelova univerzita v Brně, Zemědělská 1, 613 00 Brno, Česká republika,
E-mailová adresa: rybicka@mendelu.cz, <http://akela.mendelu.cz/~rybicka>