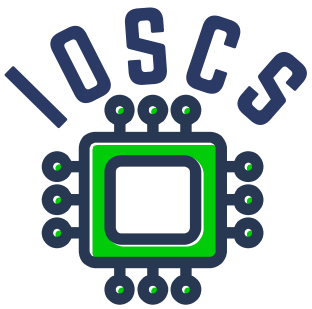
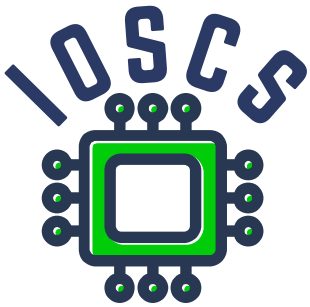


Project: Innovative Open Source Courses for Computer Science

Open source nástroje pro zpracování textů
Prezentace

Jiří Rybička
Mendelova univerzita v Brně

29. 5. 2021



This teaching material was written as one of the outputs of the project "Innovative Open Source Courses for Computer Science", funded by the Erasmus+ grant no. 2019-1-PL01-KA203-065564. The project is coordinated by West Pomeranian University of Technology in Szczecin (Poland) and is implemented in partnership with Mendel University in Brno (Czech Republic) and University of Žilina (Slovak Republic). The project implementation timeline is September 2019 to December 2022.

Project information

Project was implemented under the Erasmus+.
Project name: **"Innovative Open Source courses for Computer Science curriculum"**
Project nr: **2019-1-PL01-KA203-065564**
Key Action: **KA2 – Cooperation for innovation and the exchange of good practices**
Action Type: **KA203 – Strategic Partnerships for higher education**

Consortium
ZACHODNIOPOMORSKI UNIWERSYTET TECHNOLOGICZNY W SZCZECINIE
MENDELOVA UNIVERZITA V BRNĚ
ŽILINSKÁ UNIVERZITA V ŽILINE

Erasmus+ Disclaimer
This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Copyright Notice
This content was created by the IOSCS consortium: 2019–2022. The content is copyrighted and distributed under Creative Commons Attribution-ShareAlike 4.0 International licence (CC BY-SA 4.0).

Co-funded by the
Erasmus+ Programme
of the European Union

Metody zpracování dokumentů

Open source tools for text processing

Jiří Rybička
Department of Informatics
FBE MENDELU in Brno
rybicka@mendelu.cz

Project: Innovative Open Source Courses for Computer Science

Funded by
the European Union

Obsah

Prvky dokumentů
TEX
Hladká sazba

- Nový přístup ke zpracování dokumentů

Obsah

Prvky dokumentů
TEX
Hladká sazba

- Nový přístup ke zpracování dokumentů
- Typografie jako druhý krok

Obsah

Prvky dokumentů
TEX
Hladká sazba

- Nový přístup ke zpracování dokumentů
- Typografie jako druhý krok
- Strukturní značkování jako obecný nástroj

Obsah

Prvky dokumentů
TEX
Hladká sazba

- Nový přístup ke zpracování dokumentů
- Typografie jako druhý krok
- Strukturní značkování jako obecný nástroj
- Implementace dokumentů pomocí open source nástrojů

Dokument

- Dokument je spojení obsahu a formy

Dokument

- Dokument je spojení obsahu a formy
- Autor – Úpravce – Sazeč

Dokument

- Dokument je spojení obsahu a formy
- Autor – Úpravce – Sazeč
- Rozpoznání prvků v dokumentu

Dokument

- Dokument je spojení obsahu a formy
- Autor – Úpravce – Sazeč
- Rozpoznání prvků v dokumentu
- Vizuální podoba prvků dokumentu – typografie

Implementace

- Strukturní značkování

Implementace

- Strukturní značkování
- Oddělení definice značkování od vlastního dokumentu

Implementace

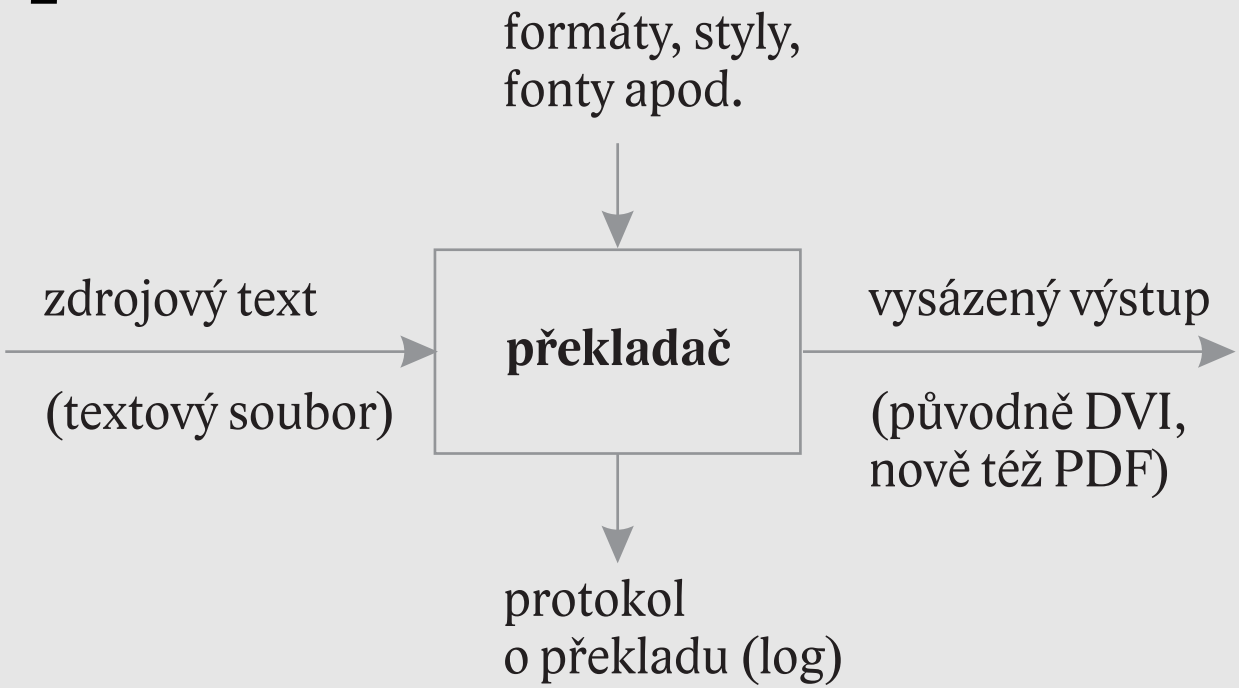
- Strukturní značkování
- Oddělení definice značkování od vlastního dokumentu
- Možnosti strukturního značkování v různých systémech

Implementace

- Strukturní značkování
- Oddělení definice značkování od vlastního dokumentu
- Možnosti strukturního značkování v různých systémech
- Nástroje open source pro zpracování textů

Prvky dokumentů
TEX
Hladká sazba

- Základní princip funkce systémů založených na TEXu



Prvky dokumentů
TEX
Hladká sazba

- Stručná historie TEXu

Prvky dokumentů
TEX
Hladká sazba

- Stručná historie TEXu
- Rozšíření (LATEX, XETEX, XELATEX), distribuce

Prvky dokumentů
TEX
Hladká sazba

- Stručná historie TEXu
- Rozšíření (LATEX, XETEX, XELATEX), distribuce
- TEXonWeb, Overleaf

- Stručná historie TEXu
- Rozšíření (LATEX, XETEX, XELATEX), distribuce
- TEXonWeb, Overleaf
- Makrojazyk TEXu: aktivní znaky, příkazy, makra

- Stručná historie TEXu
- Rozšíření (LATEX, XETEX, XELATEX), distribuce
- TEXonWeb, Overleaf
- Makrojazyk TEXu: aktivní znaky, příkazy, makra
- XELATEX: příkazy, parametry, rozsah platnosti (skupiny, prostředí)

TEX

- Stručná historie TEXu
- Rozšíření (LATEX, XETEX, XELATEX), distribuce
- TEXonWeb, Overleaf
- Makrojazyk TEXu: aktivní znaky, příkazy, makra
- XELATEX: příkazy, parametry, rozsah platnosti (skupiny, prostředí)
- Možnosti definice nových příkazů

- Stručná historie TEXu
- Rozšíření (LATEX, XETEX, XELATEX), distribuce
- TEXonWeb, Overleaf
- Makrojazyk TEXu: aktivní znaky, příkazy, makra
- XELATEX: příkazy, parametry, rozsah platnosti (skupiny, prostředí)
- Možnosti definice nových příkazů
- Implementace dokumentu, styly a definice strukturního značkování

- Můj první dokument (přehled), práce s TEXonWeb
  `tex.mendelu.cz/new`; `tex.mendelu.cz/new/auth`

- Můj první dokument (přehled), práce s TEXonWeb
  `tex.mendelu.cz/new`; `tex.mendelu.cz/new/auth`
- Styly (předdefinované, uživatelské)

- Můj první dokument (přehled), práce s TEXonWeb
  `tex.mendelu.cz/new`; `tex.mendelu.cz/new/auth`
- Styly (předdefinované, uživatelské)
- Definice nových příkazů (makra) – základy

- Můj první dokument (přehled), práce s TEXonWeb
  `tex.mendelu.cz/new`; `tex.mendelu.cz/new/auth`
- Styly (předdefinované, uživatelské)
- Definice nových příkazů (makra) – základy
- Překlad, logový soubor, chyby

Základní písmo

- Typy písma: strojopisné/knižní; 3 kategorie

Základní písmo

- Typy písma: strojopisné/knižní; 3 kategorie
- Serifová písma – základní text v tištěných dokumentech

Základní písmo

- Typy písma: strojopisné/knižní; 3 kategorie
- Serifová písma – základní text v tištěných dokumentech
- Bezserifová písma – druhý doplňkový font v tištěných dokumentech, primární font v elektronických dokumentech

Základní písmo

- Typy písma: strojopisné/knižní; 3 kategorie
- Serifová písma – základní text v tištěných dokumentech
- Bezserifová písma – druhý doplňkový font v tištěných dokumentech, primární font v elektronických dokumentech
- Jiná písma: příležitostné tištěné nebo elektronické dokumenty, jak například pozvánky, oznámení, reklamy

Základní písmo

- Typy písma: strojopisné/knižní; 3 kategorie
- Serifová písma – základní text v tištěných dokumentech
- Bezserifová písma – druhý doplňkový font v tištěných dokumentech, primární font v elektronických dokumentech
- Jiná písma: příležitostné tištěné nebo elektronické dokumenty, jak například pozvánky, oznámení, reklamy
- Optimální řešení: jeden dokument – jeden typ písma

Základní písmo

- Typy písma: strojopisné/knižní; 3 kategorie
- Serifová písma – základní text v tištěných dokumentech
- Bezserifová písma – druhý doplňkový font v tištěných dokumentech, primární font v elektronických dokumentech
- Jiná písma: příležitostné tištěné nebo elektronické dokumenty, jak například pozvánky, oznámení, reklamy
- Optimální řešení: jeden dokument – jeden typ písma
- Míchání typů písma: základní text je serifový, ale hlavičky, titulky apod. bezserifové

Základní písmo – velikosti

- Stupeň (velikost) písma: parametr odvozený z kovové sazby

Základní písmo – velikosti

- Stupeň (velikost) písma: parametr odvozený z kovové sazby
- Měrné systémy: základní jednotkou je **typografický bod (point) – pt**

Základní písmo – velikosti

- Stupeň (velikost) písma: parametr odvozený z kovové sazby
- Měrné systémy: základní jednotkou je **typografický bod (point) – pt**
- Anglický systém: 1 pt = 0,353 mm

Základní písmo – velikosti

- Stupeň (velikost) písma: parametr odvozený z kovové sazby
- Měrné systémy: základní jednotkou je **typografický bod (point) – pt**
- Anglický systém: 1 pt = 0,353 mm
- Větší jednotka: 1 pica = 12 pt

Základní písmo – velikosti

- Stupeň (velikost) písma: parametr odvozený z kovové sazby
- Měrné systémy: základní jednotkou je **typografický bod (point) – pt**
- Anglický systém: 1 pt = 0,353 mm
- Větší jednotka: 1 pica = 12 pt
- Stupeň písma: základní text v knihách: 10–12 pt

Základní písmo – velikosti

- Stupeň (velikost) písma: parametr odvozený z kovové sazby
- Měrné systémy: základní jednotkou je **typografický bod (point) – pt**
- Anglický systém: 1 pt = 0,353 mm
- Větší jednotka: 1 pica = 12 pt
- Stupeň písma: základní text v knihách: 10–12 pt
- Jiné stupně: poznámky pod čarou 8 pt, nadpisy sekcí 12–24 pt

Hladká sazba

- Hladká sazba je část dokumentu s jedním typem, řezem a stupněm písma

Hladká sazba

- Hladká sazba je část dokumentu s jedním typem, řezem a stupněm písma
- Speciální znaky – mezery, pomlčky, spojovníky, uvozovky...

Hladká sazba

- Hladká sazba je část dokumentu s jedním typem, řezem a stupněm písma
- Speciální znaky – mezery, pomlčky, spojovníky, uvozovky...
- Každý speciální znak má svůj tvar, mezerování a umístění v textu

Hladká sazba

- Hladká sazba je část dokumentu s jedním typem, řezem a stupněm písma
- Speciální znaky – mezery, pomlčky, spojovníky, uvozovky...
- Každý speciální znak má svůj tvar, mezerování a umístění v textu
- Závisí to na použitém jazyce, některé znaky se v různých jazycích zapisují různě (např. uvozovky)

Hladká sazba

- Hladká sazba je část dokumentu s jedním typem, řezem a stupněm písma
- Speciální znaky – mezery, pomlčky, spojovníky, uvozovky...
- Každý speciální znak má svůj tvar, mezerování a umístění v textu
- Závisí to na použitém jazyce, některé znaky se v různých jazycích zapisují různě (např. uvozovky)
- Jazyková a typografická pravidla definují správný tvar a umístění

Smíšená a odstavcová sazba

Open source tools for text processing

Jiří Rybička
Department of Informatics
FBE MENDELU in Brno
rybicka@mendelu.cz

Project: Innovative Open Source Courses for Computer Science

Funded by
the European Union

Typy písma

- Typy písma ve smíšené sazbě – optimální je použít méně než 3 typy

Typy písma

- Typy písma ve smíšené sazbě – optimální je použít méně než 3 typy
- Doplňkové písmo – pro hlavičky, popisky, tabulková data atd.

Typy písma

- Typy písma ve smíšené sazbě – optimální je použít méně než 3 typy
- Doplňkové písmo – pro hlavičky, popisky, tabulková data atd.
- Obvyklá varianta pro tištěné dokumenty: základní font serifový, doplňkový bezserifový ze stejné rodiny nebo z *vizuálně kompatibilní* rodiny

Typy písma

- Typy písma ve smíšené sazbě – optimální je použít méně než 3 typy
- Doplňkové písmo – pro hlavičky, popisky, tabulková data atd.
- Obvyklá varianta pro tištěné dokumenty: základní font serifový, doplňkový bezserifový ze stejné rodiny nebo z *vizuálně kompatibilní* rodiny
- Vizuální kompatibilita: ideální řešení je speciální dvojice přímo z profesionální písmolijny (např. Baskerville + John Sans od F. Štorma)

Typy písma

- Typy písma ve smíšené sazbě – optimální je použít méně než 3 typy
- Doplňkové písmo – pro hlavičky, popisky, tabulková data atd.
- Obvyklá varianta pro tištěné dokumenty: základní font serifový, doplňkový bezserifový ze stejné rodiny nebo z *vizuálně kompatibilní* rodiny
- Vizuální kompatibilita: ideální řešení je speciální dvojice přímo z profesionální písmolijny (např. Baskerville + John Sans od F. Štorma)
- Výhody bezserifového doplňkového písma: výrazné a dobře čitelné v krátkých nápisech (řádek nadpisu, krátký popisek, stránkové záhlaví apod.)

Typy písma – implementace

- XeLaTeX: `\fontspec{type}`

Typy písma – implementace

- XeLaTeX: `\fontspec{type}`
- Je k dispozici jakýkoliv font instalovaný v systému, formáty: TTF, OTF, Adobe Type 1

Typy písma – implementace

- XeLaTeX: `\fontspec{type}`
- Je k dispozici jakýkoliv font instalovaný v systému, formáty: TTF, OTF, Adobe Type 1
- Volitelné parametry: `\fontspec[options]{type}`

Typy písma – implementace

- XeLaTeX: `\fontspec{type}`
- Je k dispozici jakýkoliv font instalovaný v systému, formáty: TTF, OTF, Adobe Type 1
- Volitelné parametry: `\fontspec[options]{type}`
- Nejužívanější parametr: `[Mapping=tex-text]` – používá automatické ligatury pro pomlčky apod. jako standardní fonty TeXu (Computer/Latin Modern)

Písmové řezy

- Řez písma – modifikace základního tvaru písma

Písmové řezy

- Řez písma – modifikace základního tvaru písma
- Modifikace ve sklonu (kurzíva, skloněné)

Písmové řezy

- Řez písma – modifikace základního tvaru písma
- Modifikace ve sklonu (kurzíva, skloněné)
- Modifikace v duktu (tenké, světlé, polotučné, tučné, těžké, tmavé)

Písmové řezy

- Řez písma – modifikace základního tvaru písma
- Modifikace ve sklonu (kurzíva, skloněné)
- Modifikace v duktu (tenké, světlé, polotučné, tučné, těžké, tmavé)
- Modifikace v šířce znaků (zúžené, rozšířené)

Písmové řezy

- Řez písma – modifikace základního tvaru písma
- Modifikace ve sklonu (kurzíva, skloněné)
- Modifikace v duktu (tenké, světlé, polotučné, tučné, těžké, tmavé)
- Modifikace v šířce znaků (zúžené, rozšířené)
- Modifikace tahů (zdobné, obrysové)

Písmové řezy

- Řez písma – modifikace základního tvaru písma
- Modifikace ve sklonu (kurzíva, skloněné)
- Modifikace v duktu (tenké, světlé, polotučné, tučné, těžké, tmavé)
- Modifikace v šířce znaků (zúžené, rozšířené)
- Modifikace tahů (zdobné, obrysové)
- Kombinace: tučná kurzíva, tučné rozšířené, světlé zúžené apod.

Písmové řezy

- Řez písma – modifikace základního tvaru písma
- Modifikace ve sklonu (kurzíva, skloněné)
- Modifikace v duktu (tenké, světlé, polotučné, tučné, těžké, tmavé)
- Modifikace v šířce znaků (zúžené, rozšířené)
- Modifikace tahů (zdobné, obrysové)
- Kombinace: tučná kurzíva, tučné rozšířené, světlé zúžené apod.
- Speciální případ: kapitálky

Jak vyznačovat

- Podle typografických pravidel vyznačujeme kurzívním řezem

Jak vyznačovat

- Podle typografických pravidel vyznačujeme kurzívním řezem
- Příkaz LaTeXu je `\emph{text}`. Příkaz je použitelný uvnitř jiného příkazu `\emph{}` a automaticky použije obyčejný řez (v souhlase s typografickými pravidly)

Jak vyznačovat

- Podle typografických pravidel vyznačujeme kurzívním řezem
- Příkaz LaTeXu je `\emph{text}`. Příkaz je použitelný uvnitř jiného příkazu `\emph{}` a automaticky použije obyčejný řez (v souhlase s typografickými pravidly)
- Změna řezu – dva způsoby: příkaz s parametrem, např. `\textbf{text}`; příkaz jako přepínač – pracuje uvnitř skupiny, např. `{\bfseries text}`

Jak vyznačovat

- Podle typografických pravidel vyznačujeme kurzívním řezem
- Příkaz LaTeXu je `\emph{text}`. Příkaz je použitelný uvnitř jiného příkazu `\emph{}` a automaticky použije obyčejný řez (v souhlase s typografickými pravidly)
- Změna řezu – dva způsoby: příkaz s parametrem, např. `\textbf{text}`; příkaz jako přepínač – pracuje uvnitř skupiny, např. `{\bfseries text}`
- Další příkazy pro změnu řezu viz učebnice

Písmové stupně

- Základní stupeň písma je nastaven na 10 pt, jiné stupně určuje volitelný parametr příkazu `\documentclass`: `[11pt]` nebo `[12pt]`

Písmové stupně

- Základní stupeň písma je nastaven na 10 pt, jiné stupně určuje volitelný parametr příkazu `\documentclass`: `[11pt]` nebo `[12pt]`
- Dokumentní prvky s různými stupni písma jsou často řešeny jako předdefinované příkazy, např. `\section` nebo `\footnote`

Písmové stupně

- Základní stupeň písma je nastaven na 10 pt, jiné stupně určuje volitelný parametr příkazu `\documentclass`: `[11pt]` nebo `[12pt]`
- Dokumentní prvky s různými stupni písma jsou často řešeny jako předdefinované příkazy, např. `\section` nebo `\footnote`
- Nastavení stupně písma je možné přepínacími příkazy s relativními velikostmi: `\tiny`; `\scriptsize`; `\footnotesize`; `\small`; `\normalsize`; `\large`; `\Large`; `\LARGE`; `\huge` a `\Huge`

Písmové stupně

- Základní stupeň písma je nastaven na 10 pt, jiné stupně určuje volitelný parametr příkazu `\documentclass`: `[11pt]` nebo `[12pt]`
- Dokumentní prvky s různými stupni písma jsou často řešeny jako předdefinované příkazy, např. `\section` nebo `\footnote`
- Nastavení stupně písma je možné přepínacími příkazy s relativními velikostmi: `\tiny`; `\scriptsize`; `\footnotesize`; `\small`; `\normalsize`; `\large`; `\Large`; `\LARGE`; `\huge` a `\Huge`
- Jakýkoliv stupeň lze nastavit příkazem `\fontsize{velikost}{řádkování}`

Písmové stupně

- Základní stupeň písma je nastaven na 10 pt, jiné stupně určuje volitelný parametr příkazu `\documentclass`: `[11pt]` nebo `[12pt]`
- Dokumentní prvky s různými stupni písma jsou často řešeny jako předdefinované příkazy, např. `\section` nebo `\footnote`
- Nastavení stupně písma je možné přepínacími příkazy s relativními velikostmi: `\tiny`; `\scriptsize`; `\footnotesize`; `\small`; `\normalsize`; `\large`; `\Large`; `\LARGE`; `\huge` a `\Huge`
- Jakýkoliv stupeň lze nastavit příkazem `\fontsize{velikost}{řádkování}`
- Příkaz musí být následován příkazem `\selectfont`

Odstavcové parametry

- Geometrické parametry (viz obrázek v učebnici): odsazení, zarážka, levý/pravý okraj, řádkování, zarovnání

Odstavcové parametry

- Geometrické parametry (viz obrázek v učebnici): odsazení, zarážka, levý/pravý okraj, řádkování, zarovnání
- Standardní chování: Odsazení 0 pt, zarážka 15 pt, levý a pravý okraj 0 pt (plná šíře sazby), řádkování 12 pt pro desetibodový stupeň textu; zarovnání do bloku

Odstavcové parametry

- Geometrické parametry (viz obrázek v učebnici): odsazení, zarážka, levý/pravý okraj, řádkování, zarovnání
- Standardní chování: Odsazení 0 pt, zarážka 15 pt, levý a pravý okraj 0 pt (plná šíře sazby), řádkování 12 pt pro desetibodový stupeň textu; zarovnání do bloku
- Parametry jsou většinou *délky*. Délky jsou uloženy v délkových registrech

Odstavcové parametry

- Geometrické parametry (viz obrázek v učebnici): odsazení, zarážka, levý/pravý okraj, řádkování, zarovnání
- Standardní chování: Odsazení 0 pt, zarážka 15 pt, levý a pravý okraj 0 pt (plná šíře sazby), řádkování 12 pt pro desetibodový stupeň textu; zarovnání do bloku
- Parametry jsou většinou *délky*. Délky jsou uloženy v délkových registrech
- Délky mohou být pevné a pružné

Délkové jednotky

- Systém TEX má jedinečný systém délkových jednotek. Zahrnuje Didôtův evropský systém (viz učebnice), anglický systém, palce, metrický systém a speciální jednotku „scaled point“, relativní jednotky em a ex.

Délkové jednotky

- Systém TEX má jedinečný systém délkových jednotek. Zahrnuje Didôtův evropský systém (viz učebnice), anglický systém, palce, metrický systém a speciální jednotku „scaled point“, relativní jednotky em a ex.
- Jména všech jednotek jsou v následující tabulce:

Délkové jednotky

- Systém TEX má jedinečný systém délkových jednotek. Zahrnuje Didôtův evropský systém (viz učebnice), anglický systém, palce, metrický systém a speciální jednotku „scaled point“, relativní jednotky em a ex.
- Jména všech jednotek jsou v následující tabulce:

jméno	*zkratka*	*poznámka*
Anglický historický bod	pt	0,351 mm
Monotypový bod (big point)	bp	0,353 mm
pica	pc	1 pc = 12 pt
Evropský Didôtův bod	dd	0,376 mm
cicero	cc	1 cc = 12 dd
palec	in	1 in = 25,4 mm
centimetr	cm	
milimetr	mm	
scaled point	sp	65 536 sp = 1 pt

Délkové registry

- Registry mohou být předdefinované a uživatelské. Použití hodnoty z registru je jednoduché – stačí zapsat jméno registru

Délkové registry

- Registry mohou být předdefinované a uživatelské. Použití hodnoty z registru je jednoduché – stačí zapsat jméno registru
- Definice délkového registru: `\newlength\newname` (pevná délka) nebo `\newdimen\newname` (pružná)

Délkové registry

- Registry mohou být předdefinované a uživatelské. Použití hodnoty z registru je jednoduché – stačí zapsat jméno registru
- Definice délkového registru: `\newlength\newname` (pevná délka) nebo `\newdimen\newname` (pružná)
- Počáteční hodnota po definici registru je 0 pt

Délkové registry

- Registry mohou být předdefinované a uživatelské. Použití hodnoty z registru je jednoduché – stačí zapsat jméno registru
- Definice délkového registru: `\newlength\newname` (pevná délka) nebo `\newdimen\newname` (pružná)
- Počáteční hodnota po definici registru je 0 pt
- Nastavení délky (pevné): `\registr=délka`; rovnítko je volitelné

Délkové registry

- Registry mohou být předdefinované a uživatelské. Použití hodnoty z registru je jednoduché – stačí zapsat jméno registru
- Definice délkového registru: `\newlength\newname` (pevná délka) nebo `\newdimen\newname` (pružná)
- Počáteční hodnota po definici registru je 0 pt
- Nastavení délky (pevné): `\registr=délka`; rovnítko je volitelné
- Nastavení délky (pružné):
  `\registr=délka plus X minus Y`, kde X a Y jsou opět délky v libovolných jednotkách

Délkové registry

- Registry mohou být předdefinované a uživatelské. Použití hodnoty z registru je jednoduché – stačí zapsat jméno registru
- Definice délkového registru: `\newlength\newname` (pevná délka) nebo `\newdimen\newname` (pružná)
- Počáteční hodnota po definici registru je 0 pt
- Nastavení délky (pevné): `\registr=délka`; rovnítko je volitelné
- Nastavení délky (pružné): `\registr=délka plus X minus Y`, kde X a Y jsou opět délky v libovolných jednotkách
- Hodnota registru může být násobena konstantou, např. `3\registr` je třikrát hodnota registru, nebo `-0.5\registr` je polovina hodnoty registru

Délkové registry

- Registry mohou být předdefinované a uživatelské. Použití hodnoty z registru je jednoduché – stačí zapsat jméno registru
- Definice délkového registru: `\newlength\newname` (pevná délka) nebo `\newdimen\newname` (pružná)
- Počáteční hodnota po definici registru je 0 pt
- Nastavení délky (pevné): `\registr=délka`; rovnítko je volitelné
- Nastavení délky (pružné): `\registr=délka plus X minus Y`, kde X a Y jsou opět délky v libovolných jednotkách
- Hodnota registru může být násobena konstantou, např. `3\registr` je třikrát hodnota registru, nebo `-0.5\registr` je polovina hodnoty registru
- Sčítání délek: `\addtolength\registr by délka`

Odstavcové nastavení

- Předdefinované registry jsou: `\parskip` (pružná), `\parindent`, `\baselineskip`, `\leftskip`, `\rightskip` (vše pevné)

Odstavcové nastavení

- Předdefinované registry jsou: `\parskip` (pružná), `\parindent`, `\baselineskip`, `\leftskip`, `\rightskip` (vše pevné)
- Změna geometrických parametrů: např.
  `\parskip=0.5\baselineskip plus 2pt minus 1pt`
  nebo
  `\parindent=2em` (relativní; `2×` aktuální stupeň písma)

Odstavcové nastavení

- Předdefinované registry jsou: `\parskip` (pružná), `\parindent`, `\baselineskip`, `\leftskip`, `\rightskip` (vše pevné)
- Změna geometrických parametrů: např.
  `\parskip=0.5\baselineskip plus 2pt minus 1pt`
  nebo
  `\parindent=2em` (relativní; `2×` aktuální stupeň písma)
- Registr `\baselineskip` však nelze takto měnit – jeho změnu provedeme redefinicí koeficientu `\baselinestretch` z implicitní hodnoty `1` na jinou, např. `\def\baselinestretch{1.3}`

Odstavcové nastavení

- Předdefinované registry jsou: `\parskip` (pružná), `\parindent`, `\baselineskip`, `\leftskip`, `\rightskip` (vše pevné)
- Změna geometrických parametrů: např.
  `\parskip=0.5\baselineskip plus 2pt minus 1pt`
  nebo
  `\parindent=2em` (relativní; `2×` aktuální stupeň písma)
- Registr `\baselineskip` však nelze takto měnit – jeho změnu provedeme redefinicí koeficientu `\baselinestretch` z implicitní hodnoty `1` na jinou, např. `\def\baselinestretch{1.3}`
- Zarovnání odstavce je řešeno třemi prostředími: `flushleft`, `flushright` a `center`

Stránkový design

Open source tools for text processing

Jiří Rybička
Department of Informatics
FBE MENDELU in Brno
rybicka@mendelu.cz

Project: Innovative Open Source Courses for Computer Science

Funded by
the European Union

Stránkový zlom

- Téměř každý dokument je rozdělen na stránky

Stránkový zlom

- Téměř každý dokument je rozdělen na stránky
- Konec stránky není dovolen v libovolném místě

Stránkový zlom

- Téměř každý dokument je rozdělen na stránky
- Konec stránky není dovolen v libovolném místě
- Nastavení švícek: `\widowpenalty=n` je „trest“ za zlom stránky za prvním řádkem odstavce

Stránkový zlom

- Téměř každý dokument je rozdělen na stránky
- Konec stránky není dovolen v libovolném místě
- Nastavení švícek: `\widowpenalty=n` je „trest“ za zlom stránky za prvním řádkem odstavce
- `\clubpenalty=n` je „trest“ za zlom stránky před posledním řádkem odstavce

Stránkový zlom

- Téměř každý dokument je rozdělen na stránky
- Konec stránky není dovolen v libovolném místě
- Nastavení švícek: `\widowpenalty=n` je „trest“ za zlom stránky za prvním řádkem odstavce
- `\clubpenalty=n` je „trest“ za zlom stránky před posledním řádkem odstavce
- *n* je celočíselná hodnota od 0 (povoleno vždy) a 10000 (nikdy)

Stránkový zlom

- Téměř každý dokument je rozdělen na stránky
- Konec stránky není dovolen v libovolném místě
- Nastavení švícek: `\widowpenalty=n` je „trest“ za zlom stránky za prvním řádkem odstavce
- `\clubpenalty=n` je „trest“ za zlom stránky před posledním řádkem odstavce
- *n* je celočíselná hodnota od 0 (povoleno vždy) a 10000 (nikdy)
- Nepodmíněný konec stránky: `\newpage` nebo `\clearpage` nebo `\cleardoublepage`

Materiál na stránce

- Celý obsah stránky je dělen na tři části: stránkové záhlaví, sazební zrcadlo a stránková pata

Materiál na stránce

- Celý obsah stránky je dělen na tři části: stránkové záhlaví, sazební zrcadlo a stránková pata
- Obecně lze stránkový design volit příkazem `\pagestyle{X}`, kde X je: `plain`, `headings`, `myheadings` nebo `empty`

Materiál na stránce

- Celý obsah stránky je dělen na tři části: stránkové záhlaví, sazební zrcadlo a stránková pata
- Obecně lze stránkový design volit příkazem `\pagestyle{X}`, kde X je: `plain`, `headings`, `myheadings` nebo `empty`
- Příkaz `\pagestyle` by měl být zapsán v preambuli a ovlivňuje všechny následující stránky

Materiál na stránce

- Celý obsah stránky je dělen na tři části: stránkové záhlaví, sazební zrcadlo a stránková pata
- Obecně lze stránkový design volit příkazem `\pagestyle{X}`, kde X je: `plain`, `headings`, `myheadings` nebo `empty`
- Příkaz `\pagestyle` by měl být zapsán v preambuli a ovlivňuje všechny následující stránky
- Pro nastavení designu pro jednotlivou stránku lze použít `\thispagestyle{X}` se stejnými hodnotami parametru

Materiál na stránce

- Celý obsah stránky je dělen na tři části: stránkové záhlaví, sazební zrcadlo a stránková pata
- Obecně lze stránkový design volit příkazem `\pagestyle{X}`, kde X je: `plain`, `headings`, `myheadings` nebo `empty`
- Příkaz `\pagestyle` by měl být zapsán v preambuli a ovlivňuje všechny následující stránky
- Pro nastavení designu pro jednotlivou stránku lze použít `\thispagestyle{X}` se stejnými hodnotami parametru
- Materiál do záhlaví lze nastavit příkazy pro nadpisy nebo explicitně `\markright{text}` nebo `\markboth{levý text}{pravý text}`

Materiál na stránce

- Celý obsah stránky je dělen na tři části: stránkové záhlaví, sazební zrcadlo a stránková pata
- Obecně lze stránkový design volit příkazem `\pagestyle{X}`, kde X je: `plain`, `headings`, `myheadings` nebo `empty`
- Příkaz `\pagestyle` by měl být zapsán v preambuli a ovlivňuje všechny následující stránky
- Pro nastavení designu pro jednotlivou stránku lze použít `\thispagestyle{X}` se stejnými hodnotami parametru
- Materiál do záhlaví lze nastavit příkazy pro nadpisy nebo explicitně `\markright{text}` nebo `\markboth{levý text}{pravý text}`
- Poznámky pod čarou se vysázejí automaticky zápisem příkazu `\footnote{text}` v textu

Vertikální a horizontální mezery

- Vertikální mezera: `\vspace{délka}` nebo `\vspace*{délka}`

Vertikální a horizontální mezery

- Vertikální mezera: `\vspace{délka}` nebo `\vspace*{délka}`
- Příkaz pracuje pouze mezi odstavci

Vertikální a horizontální mezery

- Vertikální mezera: `\vspace{délka}` nebo `\vspace*{délka}`
- Příkaz pracuje pouze mezi odstavci
- Varianta s hvězdičkou pracuje také na začátku nebo na konci stránky

Vertikální a horizontální mezery

- Vertikální mezera: `\vspace{délka}` nebo `\vspace*{délka}`
- Příkaz pracuje pouze mezi odstavci
- Varianta s hvězdičkou pracuje také na začátku nebo na konci stránky
- Horizontální mezera: `\hspace{délka}` nebo `\hspace*{délka}`

Vertikální a horizontální mezery

- Vertikální mezera: `\vspace{délka}` nebo `\vspace*{délka}`
- Příkaz pracuje pouze mezi odstavci
- Varianta s hvězdičkou pracuje také na začátku nebo na konci stránky
- Horizontální mezera: `\hspace{délka}` nebo `\hspace*{délka}`
- Varianta s hvězdičkou pracuje také na začátku nebo na konci řádku

Vertikální a horizontální mezery

- Vertikální mezera: `\vspace{délka}` nebo `\vspace*{délka}`
- Příkaz pracuje pouze mezi odstavci
- Varianta s hvězdičkou pracuje také na začátku nebo na konci stránky
- Horizontální mezera: `\hspace{délka}` nebo `\hspace*{délka}`
- Varianta s hvězdičkou pracuje také na začátku nebo na konci řádku
- Speciální délka: `\fill` má nulovou přirozenou velikost a nekonečnou roztažitelnost

Vertikální a horizontální mezery

- Vertikální mezera: `\vspace{délka}` nebo `\vspace*{délka}`
- Příkaz pracuje pouze mezi odstavci
- Varianta s hvězdičkou pracuje také na začátku nebo na konci stránky
- Horizontální mezera: `\hspace{délka}` nebo `\hspace*{délka}`
- Varianta s hvězdičkou pracuje také na začátku nebo na konci řádku
- Speciální délka: `\fill` má nulovou přirozenou velikost a nekonečnou roztažitelnost
- `\hspace{\fill}` lze zkrátit na `\hfill`

Vertikální a horizontální mezery

- Vertikální mezera: `\vspace{délka}` nebo `\vspace*{délka}`
- Příkaz pracuje pouze mezi odstavci
- Varianta s hvězdičkou pracuje také na začátku nebo na konci stránky
- Horizontální mezera: `\hspace{délka}` nebo `\hspace*{délka}`
- Varianta s hvězdičkou pracuje také na začátku nebo na konci řádku
- Speciální délka: `\fill` má nulovou přirozenou velikost a nekonečnou roztažitelnost
- `\hspace{\fill}` lze zkrátit na `\hfill`
- `\vspace{\fill}` lze zkrátit na `\vfill`

Oddíly textu

- Celý dokument může být rozdělen na menší části: sekce, podsekce

Oddíly textu

- Celý dokument může být rozdělen na menší části: sekce, podsekce
- Nadpisy různých úrovní jsou řešeny skupinou podobných příkazů

Oddíly textu

- Celý dokument může být rozdělen na menší části: sekce, podsekce
- Nadpisy různých úrovní jsou řešeny skupinou podobných příkazů
- `\section{text}` je nejvyšší nadpis v dokumentní třídě `article`

Oddíly textu

- Celý dokument může být rozdělen na menší části: sekce, podsekce
- Nadpisy různých úrovní jsou řešeny skupinou podobných příkazů
- `\section{text}` je nejvyšší nadpis v dokumentní třídě `article`
- `\chapter{text}` je nejvyšší nadpis v dokumentní třídě `book` a `report`

Oddíly textu

- Celý dokument může být rozdělen na menší části: sekce, podsekce
- Nadpisy různých úrovní jsou řešeny skupinou podobných příkazů
- `\section{text}` je nejvyšší nadpis v dokumentní třídě `article`
- `\chapter{text}` je nejvyšší nadpis v dokumentní třídě `book` a `report`
- Další úrovně: `\subsection{}`; `\subsubsection{}`; `\paragraph{}` a `\subparagraph{}`

Oddíly textu

- Celý dokument může být rozdělen na menší části: sekce, podsekce
- Nadpisy různých úrovní jsou řešeny skupinou podobných příkazů
- `\section{text}` je nejvyšší nadpis v dokumentní třídě `article`
- `\chapter{text}` je nejvyšší nadpis v dokumentní třídě `book` a `report`
- Další úrovně: `\subsection{}`; `\subsubsection{}`; `\paragraph{}` a `\subparagraph{}`
- Každý nadpis oddílu řeší 4 úlohy: a) vizuální podobu nadpisu; b) číslování nadpisu; c) materiál do stránkového záhlaví; d) materiál do obsahu dokumentu

Varianty nadpisů s hvězdičkou

- Každý nadpisový příkaz má variantu s hvězdičkou – zde se řeší pouze vizuální podoba nadpisu

Varianty nadpisů s hvězdičkou

- Každý nadpisový příkaz má variantu s hvězdičkou – zde se řeší pouze vizuální podoba nadpisu
- Číslování může být vyřešeno manipulací s odpovídajícím čítačem (viz dále)

Varianty nadpisů s hvězdičkou

- Každý nadpisový příkaz má variantu s hvězdičkou – zde se řeší pouze vizuální podoba nadpisu
- Číslování může být vyřešeno manipulací s odpovídajícím čítačem (viz dále)
- Materiál do stránkového záhlaví lze nastavit příkazy `\markright` nebo `\markboth`

Varianty nadpisů s hvězdičkou

- Každý nadpisový příkaz má variantu s hvězdičkou – zde se řeší pouze vizuální podoba nadpisu
- Číslování může být vyřešeno manipulací s odpovídajícím čítačem (viz dále)
- Materiál do stránkového záhlaví lze nastavit příkazy `\markright` nebo `\markboth`
- Materiál do obsahu lze nastavit příkazem

```
\addcontentsline{soubor}{úroveň}{text}
```

Varianty nadpisů s hvězdičkou

- Každý nadpisový příkaz má variantu s hvězdičkou – zde se řeší pouze vizuální podoba nadpisu
- Číslování může být vyřešeno manipulací s odpovídajícím čítačem (viz dále)
- Materiál do stránkového záhlaví lze nastavit příkazy `\markright` nebo `\markboth`
- Materiál do obsahu lze nastavit příkazem `\addcontentsline{soubor}{úroveň}{text}`
- Soubor (rozšíření souboru) může být `toc` pro standardní obsah nebo `lof` pro standardní seznam obrázků nebo `lot` pro standardní seznam tabulek

Varianty nadpisů s hvězdičkou

- Každý nadpisový příkaz má variantu s hvězdičkou – zde se řeší pouze vizuální podoba nadpisu
- Číslování může být vyřešeno manipulací s odpovídajícím čítačem (viz dále)
- Materiál do stránkového záhlaví lze nastavit příkazy `\markright` nebo `\markboth`
- Materiál do obsahu lze nastavit příkazem `\addcontentsline{soubor}{úroveň}{text}`
- Soubor (rozšíření souboru) může být `toc` pro standardní obsah nebo `lof` pro standardní seznam obrázků nebo `lot` pro standardní seznam tabulek
- Úroveň může být `section`, `subsection` atd.

Varianty nadpisů s hvězdičkou

- Každý nadpisový příkaz má variantu s hvězdičkou – zde se řeší pouze vizuální podoba nadpisu
- Číslování může být vyřešeno manipulací s odpovídajícím čítačem (viz dále)
- Materiál do stránkového záhlaví lze nastavit příkazy `\markright` nebo `\markboth`
- Materiál do obsahu lze nastavit příkazem `\addcontentsline{soubor}{úroveň}{text}`
- Soubor (rozšíření souboru) může být `toc` pro standardní obsah nebo `lof` pro standardní seznam obrázků nebo `lot` pro standardní seznam tabulek
- Úroveň může být `section`, `subsection` atd.
- Libovolný další materiál do obsahu lze vložit příkazem `\addtocontents{soubor}{text}`

Číslování

- Každé číslování je spojeno s **čítačem**

Číslování

- Každé číslování je spojeno s **čítačem**
- Čítač je proměnná pro celočíselné hodnoty

Číslování

- Každé číslování je spojeno s **čítačem**
- Čítač je proměnná pro celočíselné hodnoty
- Čítače jsou předdefinované i uživatelské

Číslování

- Každé číslování je spojeno s **čítačem**
- Čítač je proměnná pro celočíselné hodnoty
- Čítače jsou předdefinované i uživatelské
- Manipulace s čítači: nastavení hodnoty, zobrazení hodnoty, sčítání, krokování, krokování s nastavením návěští, použití hodnoty ve výrazech

Číslování

- Každé číslování je spojeno s **čítačem**
- Čítač je proměnná pro celočíselné hodnoty
- Čítače jsou předdefinované i uživatelské
- Manipulace s čítači: nastavení hodnoty, zobrazení hodnoty, sčítání, krokování, krokování s nastavením návěští, použití hodnoty ve výrazech
- Předdefinované čítače jsou obvykle spojeny s některými příkazy, např. `page` pro číslování stránek, `footnote` pro číslování poznámek, `section` pro číslování sekcí

Číslování

- Každé číslování je spojeno s **čítačem**
- Čítač je proměnná pro celočíselné hodnoty
- Čítače jsou předdefinované i uživatelské
- Manipulace s čítači: nastavení hodnoty, zobrazení hodnoty, sčítání, krokování, krokování s nastavením návěští, použití hodnoty ve výrazech
- Předdefinované čítače jsou obvykle spojeny s některými příkazy, např. `page` pro číslování stránek, `footnote` pro číslování poznámek, `section` pro číslování sekcí
- Zobrazení hodnoty čítače je možné příkazem `\the`čítač, např. `\thepage` nebo `\thesection`

Manipulace s čítači

- Definice vlastního čítače: `\newcounter{jmeno}`

Manipulace s čítači

- Definice vlastního čítače: `\newcounter{jmeno}`
- Automaticky se vytvoří odpovídající příkaz `\thejmeno`

Manipulace s čítači

- Definice vlastního čítače: `\newcounter{jmeno}`
- Automaticky se vytvoří odpovídající příkaz `\thejmeno`
- Počáteční hodnota čítače je nula

Manipulace s čítači

- Definice vlastního čítače: `\newcounter{jmeno}`
- Automaticky se vytvoří odpovídající příkaz `\thejmeno`
- Počáteční hodnota čítače je nula
- Nastavení hodnoty: `\setcounter{jmeno}{hodnota}`

Manipulace s čítači

- Definice vlastního čítače: `\newcounter{jmeno}`
- Automaticky se vytvoří odpovídající příkaz
  `\thejmeno`
- Počáteční hodnota čítače je nula
- Nastavení hodnoty: `\setcounter{jmeno}{hodnota}`
- Přičtení hodnoty k čítači:
  `\addtocounter{jmeno}{hodnota}`

Manipulace s čítači

- Definice vlastního čítače: `\newcounter{jmeno}`
- Automaticky se vytvoří odpovídající příkaz
  `\thejmeno`
- Počáteční hodnota čítače je nula
- Nastavení hodnoty: `\setcounter{jmeno}{hodnota}`
- Přičtení hodnoty k čítači:
  `\addtocounter{jmeno}{hodnota}`
- Krokování po 1: `\stepcounter{jmeno}`

Manipulace s čítači

- Definice vlastního čítače: `\newcounter{jmeno}`
- Automaticky se vytvoří odpovídající příkaz `\thejmeno`
- Počáteční hodnota čítače je nula
- Nastavení hodnoty: `\setcounter{jmeno}{hodnota}`
- Přičtení hodnoty k čítači: `\addtocounter{jmeno}{hodnota}`
- Krokování po 1: `\stepcounter{jmeno}`
- Příkaz `\refstepcounter{jmeno}` krokuje po 1 a nastaví návěští na novou hodnotu čítače (používá se pro křížové odkazy)

Varianty zobrazení čítače

- Každý čítač může být použit (zobrazen) kdekoliv v dokumentu

Varianty zobrazení čítače

- Každý čítač může být použit (zobrazen) kdekoliv v dokumentu
- Příkaz `\thejmeno` (bez parametrů) vloží výstupní tvar hodnoty čítače do textu

Varianty zobrazení čítače

- Každý čítač může být použit (zobrazen) kdekoliv v dokumentu
- Příkaz `\thejmeno` (bez parametrů) vloží výstupní tvar hodnoty čítače do textu
- Výstupní tvar může být změněn redefinicí příkazu `\thejmeno`

Varianty zobrazení čítače

- Každý čítač může být použit (zobrazen) kdekoliv v dokumentu
- Příkaz `\thejmeno` (bez parametrů) vloží výstupní tvar hodnoty čítače do textu
- Výstupní tvar může být změněn redefinicí příkazu `\thejmeno`
- Možné výstupní tvary jsou: `\arabic{čítač}` (implicitně); `\alph{}` (malá písmena); `\Alph{}` (velká písmena); `\roman{}` (římské číslice malými písmeny); `\Roman{}` (římské číslice velkými písmeny); `\fnsymbol{}` (poznámkové symboly)

Varianty zobrazení čítače

- Každý čítač může být použit (zobrazen) kdekoliv v dokumentu
- Příkaz `\thejmeno` (bez parametrů) vloží výstupní tvar hodnoty čítače do textu
- Výstupní tvar může být změněn redefinicí příkazu `\thejmeno`
- Možné výstupní tvary jsou: `\arabic{čítač}` (implicitně); `\alph{}` (malá písmena); `\Alph{}` (velká písmena); `\roman{}` (římské číslice malými písmeny); `\Roman{}` (římské číslice velkými písmeny); `\fnsymbol{}` (poznámkové symboly)
- Příklad: `\def\thesection{\Roman{section}}` předefinuje arabské číslice sekcí na římské psané velkými písmeny

Závislosti čítačů

- Jeden čítač může být závislý na jiném čítači. Při krokování nadřízeného čítače se všechny závislé čítače vynulují

Závislosti čítačů

- Jeden čítač může být závislý na jiném čítači. Při krokování nadřízeného čítače se všechny závislé čítače vynulují
- Jiná metoda než krokování nadřízeného čítače nezpůsobí vynulování závislého čítače (`\addtocounter`, `\setcounter`)

Závislosti čítačů

- Jeden čítač může být závislý na jiném čítači. Při krokování nadřízeného čítače se všechny závislé čítače vynulují
- Jiná metoda než krokování nadřízeného čítače nezpůsobí vynulování závislého čítače (`\addtocounter`, `\setcounter`)
- Definice závislosti: `\newcounter{jmeno}[nadrizeny]` – nový čítač `jmeno` bude závislý na již existujícím čítači `nadrizeny`

Závislosti čítačů

- Jeden čítač může být závislý na jiném čítači. Při krokování nadřízeného čítače se všechny závislé čítače vynulují
- Jiná metoda než krokování nadřízeného čítače nezpůsobí vynulování závislého čítače (`\addtocounter`, `\setcounter`)
- Definice závislosti: `\newcounter{jmeno}[nadrizeny]` – nový čítač `jmeno` bude závislý na již existujícím čítači `nadrizeny`
- Vyjádření závislosti ve výstupní hodnotě: například `\def\thejmeno{\thenadrizeny:\arabic{jmeno}}` nastaví zobrazení hodnoty závislého čítače s okamžitou hodnotou nadřízeného čítače oddělenou dvojtečkou

Sazba matematiky

Open source tools for text processing

Jiří Rybička
Department of Informatics
FBE MENDELU in Brno
rybicka@mendelu.cz

Project: Innovative Open Source Courses for Computer Science

Funded by
the European Union

Matematika

- Pravidla pro sazbu matematiky jsou přísnější než pravidla pro text

Matematika

- Pravidla pro sazbu matematiky jsou přísnější než pravidla pro text
- Matematika má mnoho různých symbolů a každý z nich má svůj vlastní tvar, mezerování a metodu vkládání do výrazu

Matematika

- Pravidla pro sazbu matematiky jsou přísnější než pravidla pro text
- Matematika má mnoho různých symbolů a každý z nich má svůj vlastní tvar, mezerování a metodu vkládání do výrazu
- Každý symbol musí mít stejný tvar ve vysazené rovnici i v textu odstavce, může se lišit pouze ve velikosti

Matematika

- Pravidla pro sazbu matematiky jsou přísnější než pravidla pro text
- Matematika má mnoho různých symbolů a každý z nich má svůj vlastní tvar, mezerování a metodu vkládání do výrazu
- Každý symbol musí mít stejný tvar ve vysazené rovnici i v textu odstavce, může se lišit pouze ve velikosti
- Všechny matematické symboly musí být vloženy v některém matematickém prostředí

Matematické prvky
Matematická prostředí
Matematické symboly a prvky

Matematika

- Pravidla pro sazbu matematiky jsou přísnější než pravidla pro text
- Matematika má mnoho různých symbolů a každý z nich má svůj vlastní tvar, mezerování a metodu vkládání do výrazu
- Každý symbol musí mít stejný tvar ve vysazené rovnici i v textu odstavce, může se lišit pouze ve velikosti
- Všechny matematické symboly musí být vloženy v některém matematickém prostředí
- TEX a jeho rozšíření mají širokou podporu matematické sazby; je těžké najít jiný systém, který by matematiku podporoval lépe

Matematika

- Pravidla pro sazbu matematiky jsou přísnější než pravidla pro text
- Matematika má mnoho různých symbolů a každý z nich má svůj vlastní tvar, mezerování a metodu vkládání do výrazu
- Každý symbol musí mít stejný tvar ve vysazené rovnici i v textu odstavce, může se lišit pouze ve velikosti
- Všechny matematické symboly musí být vloženy v některém matematickém prostředí
- TEX a jeho rozšíření mají širokou podporu matematické sazby; je těžké najít jiný systém, který by matematiku podporoval lépe
- Říká se, že matematická sazba byl hlavní důvod pro vývoj TEXu (Knuth)

Matematická prostředí

- Pro matematické výrazy existují dva způsoby sazby: **textová matematika** (uvnitř odstavce) nebo **vysazená matematika** (mezi odstavci)

Matematická prostředí

- Pro matematické výrazy existují dva způsoby sazby: **textová matematika** (uvnitř odstavce) nebo **vysazená matematika** (mezi odstavci)
- TEX má proto dva nástroje: `$...$` (textová) a `$$...$$` (vysazená matematika)

Matematická prostředí

- Pro matematické výrazy existují dva způsoby sazby: **textová matematika** (uvnitř odstavce) nebo **vysazená matematika** (mezi odstavci)
- TEX má proto dva nástroje: `$...$` (textová) a `$$...$$` (vysazená matematika)
- LATEX přidává dvě další prostředí a přidává variantní zápisy pro uvedená dvě základní prostředí

Matematická prostředí

- Pro matematické výrazy existují dva způsoby sazby: **textová matematika** (uvnitř odstavce) nebo **vysazená matematika** (mezi odstavci)
- TEX má proto dva nástroje: `$...$` (textová) a `$$...$$` (vysazená matematika)
- LATEX přidává dvě další prostředí a přidává variantní zápisy pro uvedená dvě základní prostředí
- Textová matematika může být zapsána `\(` a `\)` nebo `\begin{math}...\end{math}`

Matematická prostředí

- Pro matematické výrazy existují dva způsoby sazby: **textová matematika** (uvnitř odstavce) nebo **vysazená matematika** (mezi odstavci)
- TEX má proto dva nástroje: `$...$` (textová) a `$$...$$` (vysazená matematika)
- LATEX přidává dvě další prostředí a přidává variantní zápisy pro uvedená dvě základní prostředí
- Textová matematika může být zapsána `\(` a `\)` nebo `\begin{math}...\end{math}`
- Vysazená matematika může být zapsána `\[` a `\]` nebo `\begin{displaymath}...\end{displaymath}`

Matematická prostředí LaTeXu

- Rozšiřující prostředí LaTeXu jsou `equation` a `eqnarray`

Matematická prostředí LaTeXu

- Rozšiřující prostředí LaTeXu jsou `equation` a `eqnarray`
- Prostředí `\begin{equation}...\end{equation}` čísluje vysazenou matematiku

Matematická prostředí LaTeXu

- Rozšiřující prostředí LaTeXu jsou `equation` a `eqnarray`
- Prostředí `\begin{equation}...\end{equation}` čísluje vysazenou matematiku
- S prostředím je spojen čítač `equation`

Matematická prostředí LaTeXu

- Rozšiřující prostředí LaTeXu jsou `equation` a `eqnarray`
- Prostředí `\begin{equation}...\end{equation}` čísluje vysazenou matematiku
- S prostředím je spojen čítač `equation`
- Čítač je automaticky krokován s umístěním každého prostředí do dokumentu; hodnota tohoto čítače může být křížově odkazována

Matematická prostředí LaTeXu

- Rozšiřující prostředí LaTeXu jsou `equation` a `eqnarray`
- Prostředí `\begin{equation}...\end{equation}` čísluje vysazenou matematiku
- S prostředím je spojen čítač `equation`
- Čítač je automaticky krokován s umístěním každého prostředí do dokumentu; hodnota tohoto čítače může být křížově odkazována
- Příklad:

  `\begin{equation} c^2= a^2+b^2\end{equation}` dává

$$c^2 = a^2 + b^2 \tag{1}$$

Matematická prostředí LaTeXu

- Prostředí `eqnarray` je určeno pro soustavy rovnic a umožňuje svislé zarovnání tří částí

Matematická prostředí LaTeXu

- Prostředí `eqnarray` je určeno pro soustavy rovnic a umožňuje svislé zarovnání tří částí
- Jedna část je vlevo, jedna uprostřed a jedna vpravo; části jsou v zápisu odděleny znakem `&`

Matematická prostředí LaTeXu

- Prostředí `eqnarray` je určeno pro soustavy rovnic a umožňuje svislé zarovnání tří částí
- Jedna část je vlevo, jedna uprostřed a jedna vpravo; části jsou v zápisu odděleny znakem `&`
- Levá část je zarovnána vpravo, prostřední na střed a pravá doleva

Matematická prostředí LaTeXu

- Prostředí `eqnarray` je určeno pro soustavy rovnic a umožňuje svislé zarovnání tří částí
- Jedna část je vlevo, jedna uprostřed a jedna vpravo; části jsou v zápisu odděleny znakem `&`
- Levá část je zarovnána vpravo, prostřední na střed a pravá doleva
- Jednoduchý příklad:

```
\begin{eqnarray}
    c^2 & = & a^2+b^2\\
    c   & = &\sqrt{a^2+b^2}
\end{eqnarray}
```

dává

$$c^2 = a^2 + b^2 \quad (2)$$

$$c = \sqrt{a^2 + b^2} \quad (3)$$

Matematická prostředí LaTeXu

- Každá rovnice v prostředí `eqnarray` je číslována. Potlačení čísla na daném řádku lze provést příkazem `\nonumber` na konci řádku

Matematická prostředí LaTeXu

- Každá rovnice v prostředí `eqnarray` je číslována. Potlačení čísla na daném řádku lze provést příkazem `\nonumber` na konci řádku
- Příkaz `\\` dělí systém na jednotlivé řádky

Matematická prostředí LaTeXu

- Každá rovnice v prostředí `eqnarray` je číslována. Potlačení čísla na daném řádku lze provést příkazem `\nonumber` na konci řádku
- Příkaz `\\` dělí systém na jednotlivé řádky
- Příkaz `\\` má volitelný parametr jako i jinde: `\\[vzdálenost]` přidává vertikální mezeru mezi řádky

Matematická prostředí LaTeXu

- Každá rovnice v prostředí `eqnarray` je číslována. Potlačení čísla na daném řádku lze provést příkazem `\nonumber` na konci řádku
- Příkaz `\\` dělí systém na jednotlivé řádky
- Příkaz `\\` má volitelný parametr jako i jinde: `\\[vzdálenost]` přidává vertikální mezeru mezi řádky
- Potlačení jakéhokoliv číslování celé soustavy lze dosáhnout použitím prostředí `eqnarray*`

Matematická prostředí LaTeXu

- Každá rovnice v prostředí `eqnarray` je číslována. Potlačení čísla na daném řádku lze provést příkazem `\nonumber` na konci řádku
- Příkaz `\\` dělí systém na jednotlivé řádky
- Příkaz `\\` má volitelný parametr jako i jinde: `\\[vzdálenost]` přidává vertikální mezeru mezi řádky
- Potlačení jakéhokoliv číslování celé soustavy lze dosáhnout použitím prostředí `eqnarray*`
- Více informací o vertikálním zarovnání viz prostředí `array`

Matematické symboly a prvky

- Matematická pravidla: proměnné jsou sázeny matematickou kurzívou (implicitní ve všech matematických prostředích)

Matematické symboly a prvky

- Matematická pravidla: proměnné jsou sázeny matematickou kurzívou (implicitní ve všech matematických prostředích)
- Konstanty, funkce, totální diferenciál a některé další případy jsou sázeny obyčejným řezem

Matematické symboly a prvky

- Matematická pravidla: proměnné jsou sázeny matematickou kurzívou (implicitní ve všech matematických prostředích)
- Konstanty, funkce, totální diferenciál a některé další případy jsou sázeny obyčejným řezem
- Matice, vektory a podobné struktury jsou sázeny tučně

Matematické symboly a prvky

- Matematická pravidla: proměnné jsou sázeny matematickou kurzívou (implicitní ve všech matematických prostředích)
- Konstanty, funkce, totální diferenciál a některé další případy jsou sázeny obyčejným řezem
- Matice, vektory a podobné struktury jsou sázeny tučně
- Potřebujeme tedy někdy změnu řezů: `\mathrm{}` pro obyčejný řez, `\mathbf{}` pro tučné a `\mathit{}` pro matematickou kurzívu

Matematické symboly a prvky

- Matematická pravidla: proměnné jsou sázeny matematickou kurzívou (implicitní ve všech matematických prostředích)
- Konstanty, funkce, totální diferenciál a některé další případy jsou sázeny obyčejným řezem
- Matice, vektory a podobné struktury jsou sázeny tučně
- Potřebujeme tedy někdy změnu řezů: `\mathrm{}` pro obyčejný řez, `\mathbf{}` pro tučné a `\mathit{}` pro matematickou kurzívu
- Mnoho symbolů je definováno jako příkazy – jejich tvar je již správně nastaven

Matematické symboly a prvky

- Matematická pravidla: proměnné jsou sázeny matematickou kurzívou (implicitní ve všech matematických prostředích)
- Konstanty, funkce, totální diferenciál a některé další případy jsou sázeny obyčejným řezem
- Matice, vektory a podobné struktury jsou sázeny tučně
- Potřebujeme tedy někdy změnu řezů: `\mathrm{}` pro obyčejný řez, `\mathbf{}` pro tučné a `\mathit{}` pro matematickou kurzívu
- Mnoho symbolů je definováno jako příkazy – jejich tvar je již správně nastaven
- Příklad: `$$C=A(\cos\alpha+\mathrm{i}\sin\alpha)= A\mathrm{e}^{\mathrm{i}\alpha}$$`

$$C = A(\cos\alpha + \mathrm{i}\sin\alpha) = A\mathrm{e}^{\mathrm{i}\alpha}$$

Matematické prvky

- Zlomky: `\frac{X}{Y}` dává

$$\frac{X}{Y}$$

Matematické prvky

- Zlomky: `\frac{X}{Y}` dává

$$\frac{X}{Y}$$

- Odmocniny: `\sqrt[n]{xyz}` dává

$$\sqrt[n]{xyz}$$

Matematické prvky

- Zlomky: `\frac{X}{Y}` dává

$$\frac{X}{Y}$$

- Odmocniny: `\sqrt[n]{xyz}` dává

$$\sqrt[n]{xyz}$$

- Indexy a exponenty: `a_1^3-a_{2x}^{3b}` dává

$$a_1^3 - a_{2x}^{3b}$$

Matematické prvky

- Sumy, limity, integrály...:

```
\sum_{a=1}^N x_a\cdot w_a
\lim_{x\rightarrow\infty}\frac{x+3}{x-1}
\int_0^\infty f(x)\mathrm{d}x
```

$$\sum_{a=1}^{N} x_a \cdot w_a$$

$$\lim_{x \to \infty} \frac{x+3}{x-1}$$

$$\int_0^\infty f(x)\mathrm{d}x$$

Matice, velké oddělovače

- Matice je implementována jako prostředí `array` (viz prostředí `tabular`)

Matice, velké oddělovače

- Matice je implementována jako prostředí `array` (viz prostředí `tabular`)
- Různé části výrazů mohou být ohraničeny velkými oddělovači (závorky apod.)

Matice, velké oddělovače

- Matice je implementována jako prostředí `array` (viz prostředí `tabular`)
- Různé části výrazů mohou být ohraničeny velkými oddělovači (závorky apod.)
- Příkazy `\left(` a `\right)` vysázejí velké závorky kolem výrazu

Matice, velké oddělovače

- Matice je implementována jako prostředí `array` (viz prostředí `tabular`)
- Různé části výrazů mohou být ohraničeny velkými oddělovači (závorky apod.)
- Příkazy `\left(` a `\right)` vysázejí velké závorky kolem výrazu
- Jednoduchý příklad:

```
\mathbf{A}=\left( % levý velký oddělovač
\begin{array}{cc} % matice, dva sloupce na střed
a_{11} & a_{12} \\
a_{21} & a_{22}
\end{array}\right)
```

$$\mathbf{A} = \left(\begin{array}{cc} a_{11} & a_{12} \\ a_{21} & a_{22} \end{array} \right)$$

Stručný přehled dalších symbolů

- Řecká abeceda: `\alpha` α `\beta` β `\gamma` γ `\delta` δ `\omega` ω `\phi` ϕ `\varphi` φ `\Delta` Δ `\Omega` Ω ...

Stručný přehled dalších symbolů

- Řecká abeceda: `\alpha` α `\beta` β `\gamma` γ `\delta` δ `\omega` ω `\phi` ϕ `\varphi` φ `\Delta` Δ `\Omega` Ω ...
- Operátory: `\cdot` $\cdot$ `\bullet` $\bullet$ `\circ` $\circ$ `\pm` $\pm$ `\times` $\times$ `\diamond` $\diamond$ `\cap` $\cap$ `\cup` $\cup$ `\oplus` $\oplus$ `\dagger` $\dagger$ ...

Stručný přehled dalších symbolů

- Řecká abeceda: `\alpha` α `\beta` β `\gamma` γ `\delta` δ `\omega` ω `\phi` ϕ `\varphi` φ `\Delta` Δ `\Omega` Ω ...
- Operátory: `\cdot` $\cdot$ `\bullet` $\bullet$ `\circ` $\circ$ `\pm` $\pm$ `\times` $\times$ `\diamond` $\diamond$ `\cap` $\cap$ `\cup` $\cup$ `\oplus` $\oplus$ `\dagger` $\dagger$ ...
- Relace: `\leq` $\leq$ `\geq` $\geq$ `\in` $\in$ `\sim` $\sim$ `\approx` $\approx$ `\equiv` $\equiv$ `\subset` $\subset$ `\supset` $\supset$ `\ll` $\ll$ `\gg` $\gg$ ...

Stručný přehled dalších symbolů

- Řecká abeceda: `\alpha` α `\beta` β `\gamma` γ `\delta` δ `\omega` ω `\phi` ϕ `\varphi` φ `\Delta` Δ `\Omega` Ω ...
- Operátory: `\cdot` $\cdot$ `\bullet` $\bullet$ `\circ` $\circ$ `\pm` $\pm$ `\times` $\times$ `\diamond` $\diamond$ `\cap` $\cap$ `\cup` $\cup$ `\oplus` $\oplus$ `\dagger` $\dagger$ ...
- Relace: `\leq` $\leq$ `\geq` $\geq$ `\in` $\in$ `\sim` $\sim$ `\approx` $\approx$ `\equiv` $\equiv$ `\subset` $\subset$ `\supset` $\supset$ `\ll` $\ll$ `\gg` $\gg$ ...
- Šipky: `\leftarrow` $\leftarrow$ `\rightarrow` $\rightarrow$ `\Leftarrow` $\Leftarrow$ `\longleftarrow` $\longleftarrow$ `\longleftrightarrow` $\longleftrightarrow$ `\uparrow` $\uparrow$ `\mapsto` $\mapsto$ `\nearrow` $\nearrow$ `\swarrow` $\swarrow$ ...

Stručný přehled dalších symbolů

- Řecká abeceda: `\alpha` α `\beta` β `\gamma` γ `\delta` δ `\omega` ω `\phi` ϕ `\varphi` φ `\Delta` Δ `\Omega` Ω ...
- Operátory: `\cdot` $\cdot$ `\bullet` $\bullet$ `\circ` $\circ$ `\pm` $\pm$ `\times` $\times$ `\diamond` $\diamond$ `\cap` $\cap$ `\cup` $\cup$ `\oplus` $\oplus$ `\dagger` $\dagger$ ...
- Relace: `\leq` $\leq$ `\geq` $\geq$ `\in` $\in$ `\sim` $\sim$ `\approx` $\approx$ `\equiv` $\equiv$ `\subset` $\subset$ `\supset` $\supset$ `\ll` $\ll$ `\gg` $\gg$ ...
- Šipky: `\leftarrow` $\leftarrow$ `\rightarrow` $\rightarrow$ `\Leftarrow` $\Leftarrow$ `\longleftarrow` $\longleftarrow$ `\longleftrightarrow` $\longleftrightarrow$ `\uparrow` $\uparrow$ `\mapsto` $\mapsto$ `\nearrow` $\nearrow$ `\swarrow` $\swarrow$ ...
- Funkce: `\sin` $\sin$ `\ln` $\ln$ `\inf` $\inf$ `\liminf` $\liminf$ `\max` $\max$ `\dim` $\dim$ `\arctan` $\arctan$ `\gcd` $\gcd$ `\lg` $\lg$ ...

Stručný přehled dalších symbolů

- Řecká abeceda: `\alpha` α `\beta` β `\gamma` γ `\delta` δ `\omega` ω `\phi` ϕ `\varphi` φ `\Delta` Δ `\Omega` Ω ...
- Operátory: `\cdot` $\cdot$ `\bullet` $\bullet$ `\circ` $\circ$ `\pm` $\pm$ `\times` $\times$ `\diamond` $\diamond$ `\cap` $\cap$ `\cup` $\cup$ `\oplus` $\oplus$ `\dagger` $\dagger$ ...
- Relace: `\leq` $\leq$ `\geq` $\geq$ `\in` $\in$ `\sim` $\sim$ `\approx` $\approx$ `\equiv` $\equiv$ `\subset` $\subset$ `\supset` $\supset$ `\ll` $\ll$ `\gg` $\gg$ ...
- Šipky: `\leftarrow` $\leftarrow$ `\rightarrow` $\rightarrow$ `\Leftarrow` $\Leftarrow$ `\longleftarrow` $\longleftarrow$ `\longleftrightarrow` $\longleftrightarrow$ `\uparrow` $\uparrow$ `\mapsto` $\mapsto$ `\nearrow` $\nearrow$ `\swarrow` $\swarrow$ ...
- Funkce: `\sin` $\sin$ `\ln` $\ln$ `\inf` $\inf$ `\liminf` $\liminf$ `\max` $\max$ `\dim` $\dim$ `\arctan` $\arctan$ `\gcd` $\gcd$ `\lg` $\lg$ ...
- Velké oddělovače: `\{` $\{$ `\lfloor` $\lfloor$ `\rfloor` $\rfloor$ `\lceil` $\lceil$ `\rceil` $\rceil$ `\langle` $\langle$ `\rangle` $\rangle$ `\|` $\|$ `\Uparrow` $\Uparrow$ ...

Stručný přehled dalších symbolů

- Řecká abeceda: `\alpha` α `\beta` β `\gamma` γ `\delta` δ `\omega` ω `\phi` ϕ `\varphi` φ `\Delta` Δ `\Omega` Ω ...
- Operátory: `\cdot` $\cdot$ `\bullet` $\bullet$ `\circ` $\circ$ `\pm` $\pm$ `\times` $\times$ `\diamond` $\diamond$ `\cap` $\cap$ `\cup` $\cup$ `\oplus` $\oplus$ `\dagger` $\dagger$ ...
- Relace: `\leq` $\leq$ `\geq` $\geq$ `\in` $\in$ `\sim` $\sim$ `\approx` $\approx$ `\equiv` $\equiv$ `\subset` $\subset$ `\supset` $\supset$ `\ll` $\ll$ `\gg` $\gg$ ...
- Šipky: `\leftarrow` $\leftarrow$ `\rightarrow` $\rightarrow$ `\Leftarrow` $\Leftarrow$ `\longleftarrow` $\longleftarrow$ `\longleftrightarrow` $\longleftrightarrow$ `\uparrow` $\uparrow$ `\mapsto` $\mapsto$ `\nearrow` $\nearrow$ `\swarrow` $\swarrow$ ...
- Funkce: `\sin` $\sin$ `\ln` $\ln$ `\inf` $\inf$ `\liminf` $\liminf$ `\max` $\max$ `\dim` $\dim$ `\arctan` $\arctan$ `\gcd` $\gcd$ `\lg` $\lg$ ...
- Velké oddělovače: `\{` $\{$ `\lfloor` $\lfloor$ `\rfloor` $\rfloor$ `\lceil` $\lceil$ `\rceil` $\rceil$ `\langle` $\langle$ `\rangle` $\rangle$ `\|` $\|$ `\Uparrow` $\Uparrow$ ...
- Další: `\aleph` $\aleph$ `\forall` $\forall$ `\infty` ∞ `\nabla` ∇ `\surd` $\surd$ `\flat` $\flat$ `\backslash` $\backslash$ `\partial` ∂ `\clubsuit` $\clubsuit$ ...

Tabulky, obrázky

Open source tools for text processing

Jiří Rybička
Department of Informatics
FBE MENDELU in Brno
rybicka@mendelu.cz

Project: Innovative Open Source Courses for Computer Science

Funded by
the European Union

Svislé zarovnání, tabelátory

Tabulky
Figures, graphics
Plovoucí prostředí

- Pro vertikální zarovnání slouží prostředí `tabbing` nebo `tabular`

Svislé zarovnání, tabelátory

- Pro vertikální zarovnání slouží prostředí `tabbing` nebo `tabular`
- Prostředí `tabbing` modeluje tabelační zarážky

Svislé zarovnání, tabelátory

- Pro vertikální zarovnání slouží prostředí `tabbing` nebo `tabular`
- Prostředí `tabbing` modeluje tabelační zarážky
- Toto prostředí je užitečné pouze pro velmi jednoduché tabulky

Svislé zarovnání, tabelátory

- Pro vertikální zarovnání slouží prostředí `tabbing` nebo `tabular`
- Prostředí `tabbing` modeluje tabelační zarážky
- Toto prostředí je užitečné pouze pro velmi jednoduché tabulky
- Základní princip: nastavení zarážky `\=` a odkaz na zarážku `\>`

Svislé zarovnání, tabelátory

- Pro vertikální zarovnání slouží prostředí `tabbing` nebo `tabular`
- Prostředí `tabbing` modeluje tabelační zarážky
- Toto prostředí je užitečné pouze pro velmi jednoduché tabulky
- Základní princip: nastavení zarážky `\=` a odkaz na zarážku `\>`
- Malý příklad: `\begin{tabbing}`

```
Město\hspace{30mm}\= Teplota \\
New York \> 25 $^\circ$C \\
Sydney   \> $-3$ $^\circ$C
\end{tabbing}
```

Město	Teplota
New York	25 °C
Sydney	−3 °C

Prostředí tabular

- Prostředí `tabular` je určeno pro uzavřené tabulky s čarami a různými možnostmi zarovnání

Prostředí tabular

- Prostředí `tabular` je určeno pro uzavřené tabulky s čarami a různými možnostmi zarovnání
- Obecný tvar tabulky:

```
\begin{tabular}{sloupce} \hline
 tab. pole & tab. pole & ... \\ \hline
 tab. pole & tab. pole & ... \\ \hline
\end{tabular}
```

Prostředí tabular

- Prostředí `tabular` je určeno pro uzavřené tabulky s čarami a různými možnostmi zarovnání
- Obecný tvar tabulky:

```
\begin{tabular}{sloupce} \hline
 tab. pole & tab. pole & ... \\ \hline
 tab. pole & tab. pole & ... \\ \hline
\end{tabular}
```

- Sloupce jsou definovány jako seznam zarovnávacích písmen: l, c, r (vlevo, střed, vpravo) s případnou svislou čarou |

Prostředí tabular

- Prostředí `tabular` je určeno pro uzavřené tabulky s čarami a různými možnostmi zarovnání
- Obecný tvar tabulky:

```
\begin{tabular}{sloupce} \hline
 tab. pole & tab. pole & ... \\ \hline
 tab. pole & tab. pole & ... \\ \hline
\end{tabular}
```

- Sloupce jsou definovány jako seznam zarovnávacích písmen: l, c, r (vlevo, střed, vpravo) s případnou svislou čarou |
- Speciální případ: odstavcový sloupec `p{šířka}` – materiál v tabulkovém poli je formátován do bloku na danou šířku

Prostředí tabular

- Prostředí `tabular` je určeno pro uzavřené tabulky s čarami a různými možnostmi zarovnání
- Obecný tvar tabulky:

```
\begin{tabular}{sloupce} \hline
 tab. pole & tab. pole & ... \\ \hline
 tab. pole & tab. pole & ... \\ \hline
\end{tabular}
```

- Sloupce jsou definovány jako seznam zarovnávacích písmen: l, c, r (vlevo, střed, vpravo) s případnou svislou čarou |
- Speciální případ: odstavcový sloupec `p{šířka}` – materiál v tabulkovém poli je formátován do bloku na danou šířku
- Příkaz `\hline` vytváří vodorovnou čáru pod daným řádkem

Příklad tabulky

- Jednoduchá tabulka s čarami a různým zarovnáním sloupců:

```
\begin{tabular}{|r|l|c|} \hline
\bfseries Č.&\bfseries Jméno &\bfseries Univerzita\\\hline
 1 & Paweł Obłąk & ZUT, Szczecin, Polska \\
 7 & Žaneta Čižmářová & MENDELU, Brno, Česko \\
 12 & Vladimír Bôčik & ŽU, Žilina, Slovensko \\ \hline
\end{tabular}
```

Č.	Jméno	Univerzita
1	Paweł Obłąk	ZUT, Szczecin, Polska
7	Žaneta Čižmářová	MENDELU, Brno, Česko
12	Vladimír Bôčik	ŽU, Žilina, Slovensko

Obrázky, schémata

- Pro vložení obrázku lze použít dvou způsobů: vložení externího souboru nebo použití kreslicích nástrojů

Obrázky, schémata

- Pro vložení obrázku lze použít dvou způsobů: vložení externího souboru nebo použití kreslicích nástrojů
- Externí soubory jsou připravovány v odpovídajících grafických editorech, takže neexistují žádné limity možností

Obrázky, schémata

- Pro vložení obrázku lze použít dvou způsobů: vložení externího souboru nebo použití kreslicích nástrojů
- Externí soubory jsou připravovány v odpovídajících grafických editorech, takže neexistují žádné limity možností
- Externí soubory mohou být vektorové nebo rastrové

Obrázky, schémata

- Pro vložení obrázku lze použít dvou způsobů: vložení externího souboru nebo použití kreslicích nástrojů
- Externí soubory jsou připravovány v odpovídajících grafických editorech, takže neexistují žádné limity možností
- Externí soubory mohou být vektorové nebo rastrové
- Vložení souboru do dokumentu řeší příkaz

  `\includegraphics{soubor}`

Obrázky, schémata

- Pro vložení obrázku lze použít dvou způsobů: vložení externího souboru nebo použití kreslicích nástrojů
- Externí soubory jsou připravovány v odpovídajících grafických editorech, takže neexistují žádné limity možností
- Externí soubory mohou být vektorové nebo rastrové
- Vložení souboru do dokumentu řeší příkaz `\includegraphics{soubor}`
- Příkaz je definován v balíku `graphicx`, který připojíme příkazem `\usepackage` v preambuli

Obrázky, schémata

- Pro vložení obrázku lze použít dvou způsobů: vložení externího souboru nebo použití kreslicích nástrojů
- Externí soubory jsou připravovány v odpovídajících grafických editorech, takže neexistují žádné limity možností
- Externí soubory mohou být vektorové nebo rastrové
- Vložení souboru do dokumentu řeší příkaz

  `\includegraphics{soubor}`
- Příkaz je definován v balíku `graphicx`, který připojíme příkazem `\usepackage` v preambuli
- Změnu některých parametrů vkládaného souboru řeší volitelné parametry

  `\includegraphics[param]{soubor}`

Vkládaná grafika – příklady

- Formáty vkládaných souborů mohou být PDF (vektorový), JPG a PNG (rastrový)

Vkládaná grafika – příklady

- Formáty vkládaných souborů mohou být PDF (vektorový), JPG a PNG (rastrový)
- `\includegraphics[width=.3\textwidth]{logo.pdf}`

Funded by
the European Union

Vkládaná grafika – příklady

- Formáty vkládaných souborů mohou být PDF (vektorový), JPG a PNG (rastrový)
- `\includegraphics[width=.3\textwidth]{logo.pdf}`

- `\includegraphics[scale=0.05, angle=45]{logo.pdf}`

Vkládaná grafika – příklady

- Formáty vkládaných souborů mohou být PDF (vektorový), JPG a PNG (rastrový)
- `\includegraphics[width=.3\textwidth]{logo.pdf}`

- `\includegraphics[scale=0.05, angle=45]{logo.pdf}`

- `\includegraphics[viewport=0 0 450 150, clip]{logo.pdf}`

Kreslení obrázků

- Speciální prostředí LaTeXu `picture` je určeno pro kreslení jednoduchých vektorových obrázků; je přímou součástí systému LaTeX, není potřeba další balík

Kreslení obrázků

- Speciální prostředí LaTeXu `picture` je určeno pro kreslení jednoduchých vektorových obrázků; je přímou součástí systému LaTeX, není potřeba další balík
- V tomto prostředí je k dispozici několik kreslicích příkazů

Kreslení obrázků

- Speciální prostředí LaTeXu `picture` je určeno pro kreslení jednoduchých vektorových obrázků; je přímou součástí systému LaTeX, není potřeba další balík
- V tomto prostředí je k dispozici několik kreslicích příkazů
- Měřítko grafických prvků je dáno hodnotou v registru `\unitlength`; implicitně 1 pt

Kreslení obrázků

- Speciální prostředí LaTeXu `picture` je určeno pro kreslení jednoduchých vektorových obrázků; je přímou součástí systému LaTeX, není potřeba další balík
- V tomto prostředí je k dispozici několik kreslicích příkazů
- Měřítko grafických prvků je dáno hodnotou v registru `\unitlength`; implicitně 1 pt
- Příkaz `\put(X, Y){element}` vloží element na pracovní plochu na souřadnice X, Y

Kreslení obrázků

- Speciální prostředí LaTeXu `picture` je určeno pro kreslení jednoduchých vektorových obrázků; je přímou součástí systému LaTeX, není potřeba další balík
- V tomto prostředí je k dispozici několik kreslicích příkazů
- Měřítko grafických prvků je dáno hodnotou v registru `\unitlength`; implicitně 1 pt
- Příkaz `\put(X, Y){element}` vloží element na pracovní plochu na souřadnice X, Y
- Velikost pracovní plochy je dána parametry prostředí `picture`

Kreslení obrázků

- Speciální prostředí LaTeXu `picture` je určeno pro kreslení jednoduchých vektorových obrázků; je přímou součástí systému LaTeX, není potřeba další balík
- V tomto prostředí je k dispozici několik kreslicích příkazů
- Měřítko grafických prvků je dáno hodnotou v registru `\unitlength`; implicitně 1 pt
- Příkaz `\put(X, Y){element}` vloží element na pracovní plochu na souřadnice X, Y
- Velikost pracovní plochy je dána parametry prostředí `picture`
- Souřadnice a velikost pracovní plochy se nekontrolují, takže každý element může být vložen kamkoliv i mimo pracovní plochu

Příklad obrázku

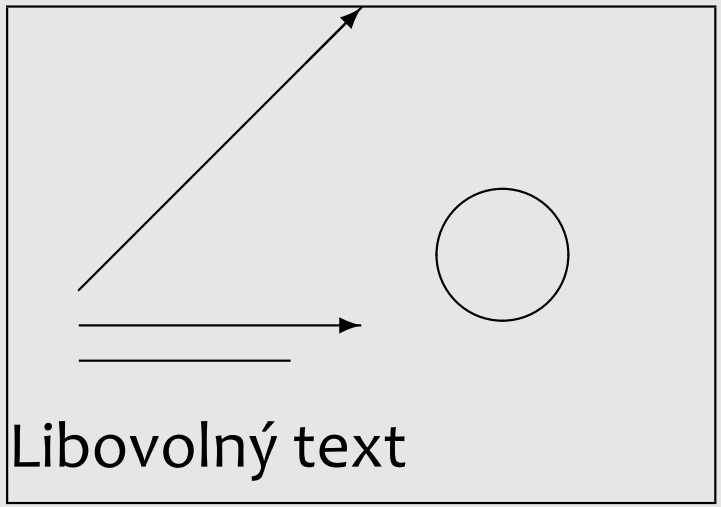
- Velikost jednotky je nastavena na 1 mm

```
\begin{picture}(100,70)
\put(0,5){Libovolný text}
\put(10,20){\line(1,0){30}}
\put(10,25){\vector(1,0){40}}
\put(10,30){\vector(1,1){40}}
\put(0,0){\framebox(100,70){}}
\put(70,35){\circle{20}}
\end{picture}
```

Příklad obrázku

- Velikost jednotky je nastavena na 1 mm

```
\begin{picture}(100,70)
\put(0,5){Libovolný text}
\put(10,20){\line(1,0){30}}
\put(10,25){\vector(1,0){40}}
\put(10,30){\vector(1,1){40}}
\put(0,0){\framebox(100,70){}}
\put(70,35){\circle{20}}
\end{picture}
```

- 

Plovoucí objekty

- Někdy není možné umístit obrázek nebo tabulku do daného místa

Plovoucí objekty

- Někdy není možné umístit obrázek nebo tabulku do daného místa
- Nevejde-li obrázek nebo tabulka na dané místo na stránce, odsune se na další a vznikne nepěkná díra

Plovoucí objekty

- Někdy není možné umístit obrázek nebo tabulku do daného místa
- Nevejde-li obrázek nebo tabulka na dané místo na stránce, odsune se na další a vznikne nepěkná díra
- K řešení tohoto problému jsou určena plovoucí prostředí

Plovoucí objekty

- Někdy není možné umístit obrázek nebo tabulku do daného místa
- Nevejde-li obrázek nebo tabulka na dané místo na stránce, odsune se na další a vznikne nepěkná díra
- K řešení tohoto problému jsou určena plovoucí prostředí
- Obsah plovoucího prostředí je vložen na nejbližší vhodné místo na následující stránce (stránkách)

Plovoucí objekty

- Někdy není možné umístit obrázek nebo tabulku do daného místa
- Nevejde-li obrázek nebo tabulka na dané místo na stránce, odsune se na další a vznikne nepěkná díra
- K řešení tohoto problému jsou určena plovoucí prostředí
- Obsah plovoucího prostředí je vložen na nejbližší vhodné místo na následující stránce (stránkách)
- Algoritmus pro vkládání plovoucích objektů lze částečně řídit uživatelskou specifikací

Plovoucí objekty

- Někdy není možné umístit obrázek nebo tabulku do daného místa
- Nevejde-li obrázek nebo tabulka na dané místo na stránce, odsune se na další a vznikne nepěkná díra
- K řešení tohoto problému jsou určena plovoucí prostředí
- Obsah plovoucího prostředí je vložen na nejbližší vhodné místo na následující stránce (stránkách)
- Algoritmus pro vkládání plovoucích objektů lze částečně řídit uživatelskou specifikací
- Existují tři plovoucí prostředí – pro tabulky, obrázky a okrajové poznámky

Plovoucí tabulky

- Existuje plovoucí prostředí `table`

Plovoucí tabulky

- Existuje plovoucí prostředí `table`
- Jednoduchý příklad:

```
\begin{table}[htbp]
\caption{Příklad plovoucí tabulky}
\begin{tabular}{|r|l|} \hline
 \bfseries Č.& \bfseries Jméno \\
 ... etc. ...
\end{tabular}\end{table}
```

Plovoucí tabulky

- Existuje plovoucí prostředí `table`
- Jednoduchý příklad:

```
\begin{table}[htbp]
\caption{Příklad plovoucí tabulky}
\begin{tabular}{|r|l|} \hline
 \bfseries Č.& \bfseries Jméno \\
 ... etc. ...
\end{tabular}\end{table}
```

- Specifikace ve volitelném parametru: h – zde (když vejde), t – vrchol stránky, b – spodek stránky, p – samostatná stránka; pořadí písmen určuje prioritu

Plovoucí tabulky

- Existuje plovoucí prostředí `table`
- Jednoduchý příklad:

```
\begin{table}[htbp]
\caption{Příklad plovoucí tabulky}
\begin{tabular}{|r|l|} \hline
 \bfseries Č.& \bfseries Jméno \\
 ... etc. ...
\end{tabular}\end{table}
```

- Specifikace ve volitelném parametru: h – zde (když vejde), t – vrchol stránky, b – spodek stránky, p – samostatná stránka; pořadí písmen určuje prioritu
- Příkaz `\caption` čísluje tabulky s připojeným čítačem `table` a umisťuje text popisku do souboru `.lot` pro seznam tabulek

Plovoucí obrázky

- Systém je podobný plovoucím tabulkám

Plovoucí obrázky

- Systém je podobný plovoucím tabulkám
- Název prostředí je `figure`

Plovoucí obrázky

- Systém je podobný plovoucím tabulkám
- Název prostředí je `figure`
- Prostředí má tentýž volitelný parametr jako `table`

Plovoucí obrázky

- Systém je podobný plovoucím tabulkám
- Název prostředí je `figure`
- Prostředí má tentýž volitelný parametr jako `table`
- Lze užít tentýž příkaz `\caption`

Plovoucí obrázky

- Systém je podobný plovoucím tabulkám
- Název prostředí je `figure`
- Prostředí má tentýž volitelný parametr jako `table`
- Lze užít tentýž příkaz `\caption`
- Číslování obrázků je řešeno čítačem `figure` a text popisku je vložen do souboru `.lof` pro seznam obrázků

Plovoucí obrázky

- Systém je podobný plovoucím tabulkám
- Název prostředí je `figure`
- Prostředí má tentýž volitelný parametr jako `table`
- Lze užít tentýž příkaz `\caption`
- Číslování obrázků je řešeno čítačem `figure` a text popisku je vložen do souboru `.lof` pro seznam obrázků
- Pořadí tabulek a pořadí obrázků není nikdy porušeno, ale obrázky a tabulky se mohou navzájem míchat