Příklady použití

Jak už bylo zmíněno v sekci pro uživatele, hlavní způsob jak ovlivnit chod programu je pozměnění vstupních souborů.

Setup JSON soubor

Jednou z možností jak pozměnit program je předefinování jeho mapování CzeTex textových funkcí na ty kódové.

Výchozím nastavení mapování je CzeTex/Setup.json, ale je předpřipravená i verze s českou lokalizací (neobsahuje všechny funkce pro demonstraci, že není potřeba implementovat všechny funkce pro fungování). Tu najdete v složce CzeTex/examples a jmenuje se nastaveni.json a soubor proNastaveni.txt využívá toto mapování.

Tedy pokud chcete vyzkoušet alternativní mapování tak můžeme příkazem:
dotnet run examples/proNastaveni.txt examples/nastaveni.json

Vytváření vstupního souboru

Projekt obsahuje desítky příkazů a nepovažuji za příznačné zde udělat vyčerpávající výčet všech, poněvadž s jiným setup souborem by byli nepoužitelné. Proto zde spíš ukážu myšlenku tvorby. Všechny příkazy potom najdete v konkrétním setup souboru, který budete chtít použít, například výchozí CzeTex/Setup.json, ve kterém jsou všechny příkazy okomentované a označené správným *hashtagem*.

Každý příkaz musí začínat speciálním znakem / a následně pokračovat názvem příkazu. Různé příkazy mají různé specifika viz dále.

Nepárové příkazy

Mezi nepárové příkazy patří mimo jiné například příkaz **/section** který vytvoří nový paragraf. Pozn: Každý text musí být umístěn v nějakém bloku nebo paragrafu.

V Setup.json jsou tyto příkazy označeny jako **#nonPairFunction**.

Párové příkazy

Mezi párové příkazy patří například vytvoření bloku tučného písma, tedy **/bold** toto je tučný text **/x** by byl tučný text. Můžete si povšimnout, že párové příkazy jsou ukončené generickým znakem **/x** z toho plyne, že se nemohou příkazy překrývat (ten, který začne později, se dříve ukončí).

V Setup.json jsou tyto příkazy označeny jako **#pairFunction**.

Příkazy s parametry

Některé příkazy mají parametry, podle kterých se mění jejich funkčnost nebo vzhled. Například funkce **/pow** má 2 parametry, základ a exponent.

Tedy pokud bychom chtěli například napsat vzorec pro obsah koule. Tak by vypadal následovně:
/math S = 4 **/times /pi /times /pow(r,2) /x** .

$$S = 4 \times \pi \times r^2$$

V Setup.json jsou tyto příkazy označeny jako **#hasParameter**.

Konkrétní příklady

Nadpis 1.úrovně

/title Toto je text nadpisu /x
nebo můžeme nahradit konkrétní nadpis generickým
/gtitle(1) Toto je text nadpisu /x

Toto je text nadpisu

Odstavec textu a speciální úpravy

/section
/bold Česko /xc plným názvem /cursive Česká republika /xc
je /underline vnitrozemský stát /x ve /linethrough východní /x
/bold /cursive střední Evropě /x /xd

Česko, plným názvem *Česká republika*, je vnitrozemský stát ve ~~východní~~ ***střední Evropě***.

Poznámka: jak můžeme vidět na příkladu, dodatečné mezery nepozmění výsledný text, ten je generován pouze podle příkazů a textu. Dokonce příkazy jako /slash, odpovídající znaku /, nevytvoří za sebou ani mezeru.

Změna nepatkového fontu na patkový (a naopak) a zarovnání textu,

/right
/serif
/size(15)
/section
Oficiálním názvem státu podle ústavy je Česká republika; jednoslovný název Česko se v ústavě nevyskytuje, je však součástí oficiální databáze OSN coby jednoslovný název státu.
/x /x /x

Oficiálním názvem státu podle ústavy je Česká republika; jednoslovný název Česko se v ústavě nevyskytuje, je však součástí oficiální databáze OSN coby jednoslovný název státu.

Poznámka: může se zdát, že obecný ukončovací příkaz dokáže být dosti nepřehledný, což se může u velmi komplikovaných výrazů stát, ale uvážil jsem, že při psaní klasického textu tato situace nenastane až tak často a napsání kratšího příkazu zvyšuje rychlost psaní textu a je pohodlnější, což jsem bral jako přednější. Navíc matematická notace, která může mít mnoho příkazů na menší části textu, má pouze jednotky párových příkazů.

Matematická notace

/section
/math
Limita f(x) jak x se přibližuje k A je rovna L /iff /forall /epsilon > 0 /exists /delta > 0 /forall x /in P(A, /delta): f(x) /in U(L, /epsilon).
/x

Limita f(x) jak x se přibližuje k A je rovna L $\iff \forall\ \varepsilon > 0\ \exists\ \delta > 0\ \forall\ x \in P(\ A, \delta\ ): f(x) \in U(\ L, \varepsilon\ )$.

Technická omezení

Při práci se seznami CzeTex umožňuje pouze globální nastavení fontu a velikosti pro celý seznam a ne pro bod nebo část bodu, speciální úpravy jako podtržení apod. nejsou podporovány vůbec.

U matematické notace je možné zadávat jako parametry u mocnin, indexů a zlomků pouze příkazy, které mají get metodu, tedy pouze speciální znaky. Není tedy možné mít zlomek ve zlomku nebo mocninu ve zlomku atd.

PRO VÍCE PŘÍKLADŮ MŮŽETE PROSTUDOVAT VZOROVÉ PŘÍKLADY V SLOŽCE CZETEX/EXAMPLES, DEFINICE FUNKCÍ V SETUP.JSON NEBO ZDROJOVÉ KÓDY K TÉTO DOKUMENTACI, PONĚVADŽ CELÁ DOKUMENTACE BYLA VYTVOŘENA POMOCÍ CZETEXU.