Dokumentace programu HTML_TeX

Václav Stupka

13. srpna 2023

Obsah

1 Úvod

HTML_TeX je jednoduchý program zajišťující překlad souborů z formátu HTML do LaTeXu. Tento projekt vznikl jako zápočtový program pro předmět Programování pro pokročilé v letním semestru prvního ročníku na Matematicko-fyzikální fakultě Univerzity Karlovy (obor Informatika). Jedná se o konzolovou aplikaci naprogramovanou v jazyce *TypeScript* a spouštěnou v prostředí *Node.js*.

2 Instalace a spuštění

Pro získání programu HTML_TeX je nutné mít nainstalované programy *Git*, *Node.js* a *npm*. Instalaci lze iniciovat těmito příkazy:

```
$ git clone git@github.com:basvas-jkj/html_TeX.git

$ cd html_TeX

$ npm install

$ node node_modules/typescript/bin/tsc
```

Poté lze program spustit příkazem `$ node dist/main.js`

3 Ovládání

Pokud je program spuštěn bez parametrů, čte standardní vstup a zapisuje na standardní výstup.

Pokud je mu předán jako parametr název souboru, čte vstup z tohoto souboru. Výstup zapisuje do souboru, jehož název získá odebráním koncovky *.html* (je-li přítomná) a přidáním koncovky *.tex*.

Pokud mu jsou předány názvy dvou souborů, čte vstup z prvního z nich a výstup zapisuje do druhého. Více než dva názvy souborů nelze zadat.

V případě, že výstupní soubor již existuje, vypíše chybu a ukončí se.

Kromě toho podporuje následující přepínače:

	význam	**delší verze**
-v	vypíše verzi a ukončí program	-version
-h	vypíše nápovědu a ukončí program	-help
-f	pokud výstupní soubor existuje, bude přepsán	-force

Dlouhou verzi je možné zadat s jedním nebo se dvěma spojovníky.

4 Struktura přeloženého dokumentu

Pro přeložený soubor ve formátu LaTeXjsem zvolil typ *article* (`\documentclass{article}`). Dále se importují následující balíčky: *graphicx*, *hyperref* a *babel* (pouze v případě, že je nastaven jazyk). Následuje vlastní obsah dokumentu. Jako první se volá příkaz `maketitle`, který vypíše titulek, jméno autora a datum vytvoření. Tyto atributy je možné nastavit v hlavičce vstupního souboru. Zbytek výstupního souboru tvoří přeložený obsah těla HTML dokumentu.

5 Postup parsování a překladu

Program HTML_TeX podporuje především základní tagy určené k formátování a strukturování textu (jejich seznam viz příští kapitola). Rozhodl jsem se neimplementovat pokročilejší tagy, jako jsou například tabulky. Opomenul jsem také všechny tagy a atributy, které souvisí s *CSS* nebo s *JavaScriptem*.

Program při zpracování vstupu postupuje podle specifikace HTML [1]. Tato specifikace však nebyla implementována kompletně. Některé vlastnosti HTML jsem vynechal, zároveň na některé situace reaguji způsobem, který více vyhovuje potřebám tohoto projektu. Zejména v případě, že HTML_TeX narazí na tag, který nezná, okamžitě se ukončí s chybou *„This tag is not supported“*. Parsovací chyby, které popisuje specifikace, jsou oznámeny hláškou *„Parse error“* a stručným popisem chyby. Parsovací chyby obvykle nezpůsobí ukončení programu, specifikace HTML totiž popisuje způsob, jak na chybu vyřešit. Můj program ho většinou respektuje.

Překlad má tři fáze: tokenizace (načtení vstupního souboru a jeho rozložení na jednotlivé prvky – otevírací a uzavírací tagy, znaky, komentáře, `DOCTYPE` a konec souboru), sestavení stromu a vlastní překlad do LaTeXu. První dvě fáze probíhají paralelně, protože se mohou navzájem ovlivňovat. Poslední fáze se spustí ve chvíli, kdy je strom kompletně sestaven.

Tokenizace i sestavení stromu představují stavový algoritmus (tj. další krok je určen nejenom vstupem, ale také stavem). Tokenizace podporuje stavy související s `DOCTYPE`, komentáři, HTML tagy a jejich atributy. Sestavení stromu podporuje tzv. módy, které souvisí s hlavičkou a tělem HTML dokumentu (tagy `<head>` a `<body>`) a s tagy, které mohou obsahovat pouze text.

Vlastní překlad je rekurzivní algoritmus, který nejprve přečte hlavičku HTML, na základě toho určí titulek, autora a jazyk dokumentu a datum jeho vytvoření. Poté se rekurzivně spustí na podstrom těla dokumentu. V této fázi se již žádná běhová chyba neobjevuje, protože předchozí dvě fáze odmítnou

cokoli, co by tento program nedokázal přeložit.

6 Podporované HTML tagy

Speciální význam v HTML má tag <`!DOCTYPE`>. Dle specifikace může mít řadu různých podob, dnes se však vyskytuje převážně podoba <`!DOCTYPE html`>. Můj program proto umožňuje pouze tuto verzi.

6.1 hlavička HTML

HTML	LaTeX	význam	výchozí hodnota
<title>	\title{...}	titulek	„empty“
<meta author>	\author{...}	autor	„unknown“
<meta date>	\date{...}	datum	není
<meta language>	\usepackage[...]{babel}	jazyk	není

(tagy <meta> se zapisují způsobem <`meta name="..." content="..."`>, v tabulce jsem kvůli úspoře místa zvolil zkrácený zápis)

6.2 strukturování textu

HTML	LaTeX	význam
<h1>	\section*{...}	nadpis první úrovně
<h2>	\subsection*{...}	nadpis druhé úrovně
<h3>	\subsubsection*{...}	nadpis třetí úrovně
<h4>	\paragraph{...}	nadpis čtvrté úrovně
<h5>	\subparagraph{...}	nadpis páté úrovně
<p>	„prázdný řádek“	odstavec
	\begin{itemize} ... \end{itemize}	odrážkový seznam
	\begin{enumerate} ... \end{enumerate}	číslovaný seznam
	\item ...	jeden záznam
<dl>	\begin{description} ... \end{description}	seznam definic
<dt>	\item[...]	definovaný pojem
<dl>	...	vlastní definice

(tag <h6>jsem vynechal, protože LaTeXový článek nepodporuje nadpis šesté úrovně)

6.3 formátování

HTML	LaTeX	význam
<b>	\textbf{...}	tučné písmo
	\textbf{...}	tučné písmo
<i>	\textit{...}	kurziva
	\emph{...}	zvýraznění
<u>	\underline{...}	podtržení
<big>	{\large ...}	velké písmo
<small>	{\small ...}	malé písmo
<code>	\texttt{...}	strojopis
<tt>	\texttt{...}	strojopis

6.4 ostatní tagy

HTML	LaTeX	význam
<a href>	\href{href}{...}	odkaz
<img src height width>	\includegraphics[height,width]{src}	obrázek
 	\\	nový řádek
<hr>	\noindent\hrule	vodorovná oddělovací čára

7 Použité technologie a knihovny

K vytvoření samotné aplikace jsem použil programovací jazyk *TypeScript* a běhové prostředí *Node.js*. Pro získání potřebných programů a knihoven jsem využil balíčkovací systém *npm*. Z externích knihoven jsem využil pouze *lodash*, která mi usnadnila porovnávání objektů.

Programoval jsem ve vývojovém prostředí *Visual Studio*, zdrojový kód mi kontroloval *ESlint*, verzování projektu mi zajistil systém *Git* a tuto dokumentaci píšu v *TexStudiu*.

8 Přehled souborů

src/main.ts Vstupní bod programu. Zajišťuje zpracování parametrů příkazové řádky. Vypisuje zachycené chybové hlášky.

src/buffer.ts Zjednodušuje čtení vstupu. Umožňuje přečíst jeden znak, vrátit zpátky přečtený znak a určit, zda se na vstupu nachází zadaná posloupnost znaků.

src/char.ts Obsahuje pomocné funkce pro určení typu znaku.

src/token.ts Představuje jednu jednotku kódu HTML v první fázi parsování.

src/tokeniser.ts Provede první fázi parsování. Rozloží kód HTML na jednotlivé tokeny a předá je souboru *src/tree.ts* do druhé fáze.

src/tree.ts Provede druhou fázi parsování. Z jednotlivých tokenů poskládá strom. Hotový strom je předán souboru *src/convertor.ts*.

src/convertor.ts Projde celý strom vytvořený souborem *src/tree.ts*, sestaví výsledný kód LaTeXu a zapíše ho na standardní výstup nebo do výstupního souboru.

ex.html Ukázkový soubor, na kterém jsem testoval funkčnost tohoto programu. Ke svému zobrazení potřebuje soubor *img.png*.

9 Možná budoucí rozšíření

Ačkoli jsem do tohoto projektu vložil nemalé množství úsilí, můj program nepokrývá HTML kompletně. Bylo by užitečné ho rozšířit o další HTML tagy, především tabulky. Kromě toho jsem vynechal také entity, které usnadňují přidávání speciálních znaků (zejména těch, které v HTML mají speciální význam).

Z dalších návrhů na vylepšení by bylo přepracovat rozhraní programu. Jeho současná podoba je velice zjednodušená, neumožňuje např. vyvolat překlad několika souborů najednou. Další zajímavou možností by byl přepínač, který by určil kódování souborů (momentálně lze překládat pouze soubory kódované v *UTF-8*, případně v kompatibilních kódováních).

Poslední úpravou, kterou by si můj program zasloužil, je lepší oznamování chyb. Většinu chybových hlášek jsem převzal přímo ze specifikace HTML [1]. Některé z těchto hlášek nemusí být plně srozumitelné. Především pouze oznamují, že daný problém nastal, ale nezmíní, na kterém řádku se nachází, ani jak daný problém napravit.

Reference

[1] Parsování HTML – specifikace.
`https://html.spec.whatwg.org/multipage/parsing.html`

[2] Zdrojový kód tohoto projektu.
`https://github.com/basvas-jkj/html_TeX`