

УДК 681.322

Г. М. Губаль

Луцький національний технічний університет

## Л<sup>A</sup>T<sub>E</sub>X ЯК ВИДАВНИЧА СИСТЕМА ДЛЯ СТВОРЕННЯ МАТЕМАТИЧНИХ ТЕКСТІВ І ДЛЯ ПРОГРАМУВАННЯ

*Губаль Г. М. Л<sup>A</sup>T<sub>E</sub>X як видавнична система для створення математичних текстів і для програмування. У статті розглядається Л<sup>A</sup>T<sub>E</sub>X як видавнична система для створення математичних текстів, її переваги і недоліки. Досліджується можливість конвертації T<sub>E</sub>X-скриптів у Microsoft Word, html-формат. Показано можливості програмування в Л<sup>A</sup>T<sub>E</sub>X.*

**Ключові слова:** текстовий редактор, текстовий процесор, видавнична система Л<sup>A</sup>T<sub>E</sub>X, конвертор, алгоритм.

**Літ. 13.**

Текстові дані є найбільш поширеним видом даних під час роботи з комп'ютером. Програми для роботи з текстами можна поділити на прості текстові процесори, потужні текстові процесори та видавничі системи.

Текстові редактори використовуються в основному для введення і редагування тексту. Вони не мають засобів для оформлення зовнішнього вигляду документа і є «примітивами», оскільки у вихідному файлі містяться тільки коди символів. Прикладами текстових редакторів для Windows є WordPad, NotePad, Блокнот; для Unix-подібних ОС – це Nano, Nedit.

Текстові процесори, на відміну від текстових редакторів, мають більше можливостей для форматування тексту, вбудовування в нього графіки, формул, таблиць і ін. Тому вони можуть бути використані не тільки для набірвання текстів, але і для створення різного роду документів, в тому числі офіційних. Класичним прикладом текстового процесора для Windows є Microsoft Word; для Unix-подібних ОС – OpenOffice.org.

Видавничі системи – це комплекс апаратних і програмних засобів, призначених для комп'ютерного набору, верстки і видання текстових та ілюстративних матеріалів. Головною відмінністю настільних видавничих систем від текстових редакторів є те, що вони призначені, в першу чергу, для оформлення складних документів (книг, журналів, дисертацій і ін.), а не для введення і редагування.

Будь-яка людина (студент або професор університету), пов'язана з математикою або іншими обчислювальними операціями стикається з проблемою вибору програми для зручного набору математичного тексту, курсової або видання нового навчального посібника відповідно, з великою кількістю формул, посилань і бібліографії. У цій ситуації важливу роль відіграють кілька факторів: зручність і швидкість роботи з комп'ютерною програмою, її функціональні можливості, які повністю задовольняють потреби, підтримка популярних форматів, якість матеріалу, що виводиться на друк.

Із першої групи факторів видно, що вибраний програмний засіб повинен не тільки дозволяти працювати з текстом, але швидко і якісно набирати та виводити різного виду математичні елементи, наприклад, формули, графіки, ілюстрації і т.д., при цьому задовольняти усім сучасним потребам з боку користувача. У даний час вибір програмних засобів для створення і редагування тексту дуже різноманітний, і задовольнити вищезгадані потреби можуть складні текстові процесори або видавничі системи. Проте можливість набору математичних елементів присутня не в кожному програмному засобі або реалізована не дуже зручно.

Прикладом текстового процесора, де дана функція присутня, але реалізована не дуже зручно для швидкого набору формул, є Microsoft Word. Звідси виникає проблема: чи є така комп'ютерна програма, яка реалізує цю функцію більш професійно?

Альтернативною заміною може бути видавнична система T<sub>E</sub>X [7], створена в 70-і роки минулого століття математиком і програмістом Стенфордського університету Дональдом Кнотом (Donald E. Knuth). T<sub>E</sub>X містить більше, ніж 900 команд і примітивів. У даний час сам T<sub>E</sub>X як видавнична система не розвивається. PlainT<sub>E</sub>X – перший макропакет, створений Дональдом Кнотом.

Л<sup>A</sup>T<sub>E</sub>X – один із найбільш популярних макропакетів на базі T<sub>E</sub>X, створений американським програмістом Леслі Лемпортом (Leslie Lamport) у 1985 році, який значно спростив і прискорив роботу з T<sub>E</sub>X. Л<sup>A</sup>T<sub>E</sub>X містить біля 600 макрокоманд. Ці команди визначають такі параметри тексту, як типи і розміри шрифтів основного тексту, заголовків, висоту і ширину сторінок,

інтервали між рядками, абзацами, відступи і багато іншого. Допускає використання більшості команд  $\text{T}_{\text{E}}\text{X}$  [7].

Фундаментальна концепція видавничої системи  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  – поняття стилю документа [3, 4, 5, 8, 9, 11, 12]. Стиль реалізується вибором базового стиля і групи коректуючих стилів, які доповнюють базовий. У даний час використовується версія  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}2_{\epsilon}$ , яка значно відрізняється від оригінальної версії Леслі Лемпорта.

$\text{Mik}_{\text{T}}\text{E}_{\text{X}}$  – це вільно розповсюджувана реалізація  $\text{T}_{\text{E}}\text{X}$  для Windows, яка містить майже всі відомі розширення.

Робота з видавничою системою  $\text{T}_{\text{E}}\text{X}$  подібна до програмування, тим більше, що ця система розрахована на використання текстового режиму, командного рядка і багатьох конфігураційних файлів.

Видавнича система  $\text{T}_{\text{E}}\text{X}$  на базі одного із макропакетів  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  актуальна для оформлення різного роду документів (навчальних посібників, підручників, дисертацій, тощо), для викладу ресурсів у платформи для організації дистанційного навчання, наприклад, в Moodle, для програмування [6].

Сформулюємо переваги і недоліки розглядуваної видавничої системи.

Переваги:

- одночасне навчання і використання не підтримується, що змушує більш серйозно підходити до процесу засвоєння системи;
- висока якість результату, недоступна іншим засобам поліграфічної підготовки (верстки) текстів;
- гнучкість. Все те, що можна уявити, може бути використано з мінімальними затратами праці (тільки професіоналом);
- простота підготовки дуже складних документів;
- велика кількість макропакетів, яка дозволяє зробити все, що в рамках “правильного” з точки зору поліграфії;
- не займає багато ресурсів у обчислювальній техніці;
- висока переносимість – один і той самий результат на будь-якій техніці;
- підтримка будь-яких мов у рамках одного документа;
- строгий підхід до оформлення утримує користувача в рамках поліграфічної культури;
- $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  представляє кілька стандартних стилів (книга, стаття, доповідь, діловий лист), за допомогою яких одержуються документи дуже високої поліграфічної якості;
- виключно просто набираються складні математичні формули (ця характеристика зробила систему  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  популярною серед науковців та інженерів);
- користувач повинен знати всього кілька команд, які легко запам'ятовуються і визначають логічну структуру документа, і майже не повинен знати про те, як документ форматується;
- більш складні елементи тексту, такі як виноски, бібліографія, зміст, список таблиць і т.п., а також прості рисунки можуть бути використані без особливих труднощів.

Недоліки:

- не є системою типу WYSIWYG: одночасне навчання і використання не підтримується, що не подобається нетерплячим. Навики набуваються і вдосконалюються тільки в процесі усвідомленого навчання. Знання обмінюється на особистий час. Серйозний підхід навіть до найбільш простого;
- потрібно володіти навиками роботи в редакторі тексту;
- орієнтована на багатозадачне середовище;
- вимагає знання елементарних основ поліграфії;
- створення нових стилів оформлення під силу лише професіоналам;
- переносимість знижується у разі використання растрової графіки.

Американська математична спільнота (AMS) має власний набір розширень, який називається  $\text{AMS-L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  і використовується для її наукових журналів [13].

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  – це єдиний формат для всіх світових наукових видань. Проте існують ситуації, в яких необхідно конвертувати  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  у Microsoft Word. Універсального конвертора з  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  у Microsoft Word не існує, тому в будь-якому разі необхідно допрацьовувати текст до прийняттого вигляду. Одні конвертори працюють краще, інші гірше, і все залежить від вихідного документа. Такими конверторами є  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}2_{\text{rtf}}$ , GrindEQ. GrindEQ конвертує Microsoft Word в  $\text{T}_{\text{E}}\text{X}$ ,  $\text{T}_{\text{E}}\text{X}$  в

Microsoft Word і MathType в Microsoft Equation. Решта конверторів в основному конвертують TEX-скрипти в html-формат, у якого немає розбиття на сторінки, а всі формули відображаються рисунками.

Умовно програми для конвертації TEX-скриптів у html-формат можна поділити на два види:

- програми, які модифікують поведінку TEX для виведення в html-формат (TEX4ht, hyperL<sup>A</sup>TEX);
- програми, які самі обробляють і перетворюють вихідний L<sup>A</sup>TEX/TEX код (L<sup>A</sup>TEX2html, HeVeA).

Програма TEX4ht визначна тим, що не вимагає яких-небудь втручань у результуючий html код, дозволяючи достатньо гнучко і майже зрозуміло конфігурувати процес перетворення і досягати необхідного результату:

- дозволяє задавати оточуючі html-теги для TEX заданих команд і оточень;
- редагувати css.

Програма TEX4ht розуміє всі команди визначені користувачем, бо є драйвером для TEX (як наприклад, pdfL<sup>A</sup>TEX). Крім того, програма TEX4ht вміє працювати з такими модулями, як makeind, multind, hyperref і т.д.

У даний час проводиться дослідна робота зі створення нової ревізії L<sup>A</sup>TEX3 видавничої системи L<sup>A</sup>TEX. Основні задачі нової версії – максимально розширити область застосування мови TEX і спростити для користувачів роботу з нею. У створенні L<sup>A</sup>TEX3 бере участь невелика група людей, які займаються роботою на волонтерській основі, не змінюючи основний принцип L<sup>A</sup>TEX – безкоштовність і доступність для усіх бажаючих.

Макропакет L<sup>A</sup>TEX може використовуватися і для програмування [1, 2, 10]. L<sup>A</sup>TEX є т'юрінг-повна мова програмування. Щоб продемонструвати можливості програмування цього макропакета розглянемо програмну реалізацію алгоритму обчислення чисел Фібоначчі.

Кожен член у послідовності чисел Фібоначчі є сумою попередніх двох членів послідовності. При цьому перші два члени послідовності чисел Фібоначчі дорівнюють одиниці.

Запишемо нову команду для обчислення цих чисел:

```
\fibonacci{10}
```

Коли ця команда викликається з документа L<sup>A</sup>TEX, вона створює список  $n$  чисел Фібоначчі (у даному прикладі  $n = 10$ ). Наведемо код для функції `\fibonacci`:

```
\newcount \temp \newcount \fone \newcount \ftwo \newcount \counter
\newcommand{\fibonacci}[1]{
\counter=#1
\fone=1
\ftwo=1
\temp=0
\the\fone, \the\ftwo
\fibloop
}

\newcommand{\fibloop}{ ,
\let\next=\fibloop
\temp=\fone
\fone=\ftwo
\advance \ftwo by \temp
\ifnum \counter \let \next=\relax \else \advance \counter by -1 \fi
\the\ftwo
\next
}
```

Спочатку ми оголошуємо змінні, які використовуються нижче. За допомогою команди `\newcount` ми маємо змінну, щоб зарезервувати місце для цілого числа. Тут ми оголошуємо змінні `\temp`, `\fone`, `\ftwo`, `\counter`. Якщо переписати ім'я однієї із змінних у програмному коді, наприклад, `\count0`, то решта сторінок документа занумеруються неправильно.

Команду `\fibonacci` ми створюємо за допомогою команди `\newcommand`, якій ми присвоюємо ім'я, наводимо число аргументів і `TeX` код для обробки аргументів. У цьому прикладі ми беремо один аргумент, кількість чисел Фібоначчі для виведення. Зміст команди такий: ми вводим початкові значення змінних, друкуємо перші два числа Фібоначчі, а тоді звертаємось до команди `\fibloop`, яка оголошується аналогічно.

Перша команда всередині команди `\fibloop` встановлює функцію `\next`, а останній рядок звертається до функції `\next`. Функція буде повторюватись, поки `\next` не зміниться внаслідок виконання коду всередині команди `\fibloop`. Щоб виконати команду `n` разів, використаємо твердження `if`, яке перевіряє значення лічильника: якщо це значення не досягло кінцевого, то його зменшуємо; а якщо значення лічильника досягло кінцевого, то встановлюємо `\next` в `\relax`, що не дасть можливість команді `\fibloop` повторюватись.

Інші команди у наведеному коді обчислюють наступне число Фібоначчі у послідовності і модифікують значення змінних, після чого вони готові для наступного проходження. Команда `\the\ftwo` друкує поточне число Фібоначчі в документ. Кома і проміжок на початку команди `\fibloop` відокремлює кожне число Фібоначчі.

Щоб побачити результат виконання наведеної програми, необхідно скопіювати вище наведений код на початок `.tex` документа і додати рядок

```
\fibonacci{n}
```

в тіло документа, замінюючи `n` числом.

1. Балдин Е. М. Компьютерная типография  $L^A T E X$ . / Е. М. Балдин. – СПб.: БХВ-Петербург, 2008.
2. Беляков Н. С.  $TeX$  для всех / Н. С. Беляков, В. Е. Палош, П. А. Садовский. – М.: Книжный дом «Либроком», 2009.
3. Дубинич В. Н. Использование системы  $L^A T E X$  для подготовки научных изданий / В. Н. Дубинич, М. В. Дубинич // Перспективы развития высшей школы: материалы V Международной науч.-метод. конф., Гродно: ГГАУ, 2012.
4. Жуков М. Ю. Оформление математических текстов при помощи пакета  $L^A T E X 2 \epsilon$  / М. Ю. Жуков, Е. В. Ширяева. – Ростов н/Д: Изд-во ООО «ЦВВР», 2003.
5. Жуков М. Ю.  $L^A T E X 2 \epsilon$ : искусство набора и вёрстки текстов с формулами / М. Ю. Жуков, Е. В. Ширяева. – Ростов н/Д: Изд-во ЮФУ, 2009.
6. Каров П. Шрифтовые технологии. Описание и инструментарий / П. Каров. Перевод с английского Карпинского О. С. и Куликова И. И. под редакцией, с предисловием и дополнением Ефимова В. В. – М.: Мир, 2001.
7. Кнут Д. Е. Всё про  $TeX$  / Д. Е. Кнут. Пер. с англ. М. В. Лисиной. – Протвино: АО  $R D T E X$ , 1993.
8. Котельников И. А.  $L^A T E X$  по-русски / И. А. Котельников, П. З. Чеботаев. – Новосибирск: Сибирский хронограф, 2004.
9. Львовский С. М. Набор и вёрстка в системе  $L^A T E X$  / С. М. Львовский. – 3-е изд., испр. и доп. – М.: МЦНМО, 2003.
10. Ширяева Е. В. Введение в  $TeX$ –программирование / Е. В. Ширяева, И. В. Ширяева. – Ростов н/Д: Изд-во ЮФУ, 2010.
11. Goossens M. The  $L^A T E X$  companion / M. Goossens, F. Mittelbach, A. Samarin. – Addison-Wesley, 1994. Русский перевод: Гуссенс М. Путеводитель по пакету  $L^A T E X$  и его расширению  $L^A T E X 2 \epsilon$  / М. Гуссенс, Ф. Миттельбах, А. Самарин. Перевод с английского Маховой О. А., Третьякова Н. В., Тюменцева Ю. В. и Чистякова В. В. под редакцией Маховой И. А. – М.: Мир, 1999.
12. Lamport L.  $L^A T E X$ . A document preparation system, user's guide and reference manual / L. Lamport. – Addison-Wesley, 1994.
13. Spivak M. The joy of  $TeX$ . A gourmet guide to typesetting with the  $A M S - T E X$  macro package. – American mathematical society, Providence, RI, 1990. Русский перевод: Спивак М. Восхитительный  $TeX$ : руководство по комфортному изготовлению научных публикаций в пакете  $A M S - T E X$  / М. Спивак. – М.: Мир, 1993.