

# TeX: первые шаги к комфорту

Алексей Хворост

Теоретические компьютерные науки

# Что такое TeX?

## Определение

TeX – набор команд для верски текста.

## Эволюция

- 1978 г. *Д. Кнут* Оригинальный пакет **TeX** (Искусство программирования);
- 1984 г. *Л. Лэмпорт* Пакет **LaTeX** (ориентированный на удобную верстку публикаций);
- 1985 г. *О. Паташник и Л. Лэмпорт* Пакет **BibTeX** (ориентированный на управление ссылками);
- 1990 г. *Г. Хэйген* Пакет **ConTeXt** (ориентированный на верстку книг);
- 1994 г. *Д. Хобби* Пакет **MetaPost** (ориентированный на создание иллюстраций);

# Что такое TeX?

## Определение

TeX – набор команд для верски текста.

## Эволюция

- 1978 г. *Д. Кнут* Оригинальный пакет **TeX** (Искусство программирования);
- 1984 г. *Л. Лэмпорт* Пакет **LaTeX** (ориентированный на удобную верстку публикаций);
- 1985 г. *О. Паташник и Л. Лэмпорт* Пакет **BibTeX** (ориентированный на управление ссылками);
- 1990 г. *Г. Хэйген* Пакет **ConTeXt** (ориентированный на верстку книг);
- 1994 г. *Д. Хобби* Пакет **MetaPost** (ориентированный на создание иллюстраций);

# Что такое TeX?

## Определение

TeX – набор команд для верски текста.

## Эволюция

- 1978 г. *Д. Кнут* Оригинальный пакет **TeX** (Искусство программирования);
- 1984 г. *Л. Лэмпорт* Пакет **LaTeX** (ориентированный на удобную верстку публикаций);
- 1985 г. *О. Паташник и Л. Лэмпорт* Пакет **BibTeX** (ориентированный на управление ссылками);
- 1990 г. *Г. Хэйген* Пакет **ConTeXt** (ориентированный на верстку книг);
- 1994 г. *Д. Хобби* Пакет **MetaPost** (ориентированный на создание иллюстраций);

# Что такое TeX?

## Определение

TeX – набор команд для верски текста.

## Эволюция

- 1978 г. *Д. Кнут* Оригинальный пакет **TeX** (Искусство программирования);
- 1984 г. *Л. Лэмпорт* Пакет **LaTeX** (ориентированный на удобную верстку публикаций);
- 1985 г. *О. Паташник и Л. Лэмпорт* Пакет **BibTeX** (ориентированный на управление ссылками);
- 1990 г. *Г. Хэйген* Пакет **ConTeXt** (ориентированный на верстку книг);
- 1994 г. *Д. Хобби* Пакет **MetaPost** (ориентированный на создание иллюстраций);

# Что такое TeX?

## Определение

TeX – набор команд для верски текста.

## Эволюция

- 1978 г. *Д. Кнут* Оригинальный пакет **TeX** (Искусство программирования);
- 1984 г. *Л. Лэмпорт* Пакет **LaTeX** (ориентированный на удобную верстку публикаций);
- 1985 г. *О. Паташник и Л. Лэмпорт* Пакет **BibTeX** (ориентированный на управление ссылками);
- 1990 г. *Г. Хэйген* Пакет **ConTeXt** (ориентированный на верстку книг);
- 1994 г. *Д. Хобби* Пакет **MetaPost** (ориентированный на создание иллюстраций);

# Что такое TeX?

## Определение

TeX – набор команд для верски текста.

## Эволюция

- 1978 г. *Д. Кнут* Оригинальный пакет **TeX** (Искусство программирования);
- 1984 г. *Л. Лэмпорт* Пакет **LaTeX** (ориентированный на удобную верстку публикаций);
- 1985 г. *О. Паташник и Л. Лэмпорт* Пакет **BibTeX** (ориентированный на управление ссылками);
- 1990 г. *Г. Хэйген* Пакет **ConTeXt** (ориентированный на верстку книг);
- 1994 г. *Д. Хобби* Пакет **MetaPost** (ориентированный на создание иллюстраций);

# С чем мы работаем на самом деле?

## Знакомые строчки

```
\usepackage[russian]{babel}  
\usepackage{amssymb}  
\usepackage{url}  
\usepackage{pgf}
```

Непонятно к чему относятся все перечисленные выше пакеты!



# С чем мы работаем на самом деле?

## Знакомые строчки

```
\usepackage[russian]{babel}  
\usepackage{amssymb}  
\usepackage{url}  
\usepackage{pgf}
```

Непонятно к чему относятся все перечисленные выше пакеты!

Знаете ли Вы какие пакеты поддерживаются Вашей сборкой TeX-a?

TeXLive (<http://tug.org/texlive/>)

- Бесплатная сборка пакетов TeX-a;
- Поддерживаются почти все пакеты (**TeX**, **LaTeX**, **BibTeX**, **ConTeXt**, **MetaPost**);
- Регулярное обновление пакетов;
- Поддержка большинства ОС (Windows, Unix);

# Какой средой разработки Вы пользуетесь для работы с TeX-ом?

- Блокнот и командная строка;
- WinEdt (а Вы знали, что это платная программа?);
- Eclipse и TeXlipse  
(<http://texlipse.sourceforge.net/release.html>);
- Kile (только для линуксойдов :));

# Какой средой разработки Вы пользуетесь для работы с TeX-ом?

- Блокнот и командная строка;
- WinEdt (а Вы знали, что это платная программа?);
- Eclipse и TeXlipse  
(<http://texlipse.sourceforge.net/release.html>);
- Kile (только для линуксойдов :));

# Какой средой разработки Вы пользуетесь для работы с TeX-ом?

- Блокнот и командная строка;
- WinEdt (а Вы знали, что это платная программа?);
- Eclipse и TeXlipse  
(<http://texlipse.sourceforge.net/release.html>);
- Kile (только для линуксойдов :));

# Какой средой разработки Вы пользуетесь для работы с TeX-ом?

- Блокнот и командная строка;
- WinEdt (а Вы знали, что это платная программа?);
- Eclipse и TeXlipse  
(<http://texlipse.sourceforge.net/release.html>);
- Kile (только для линуксойдов :));

# Какой средой разработки Вы пользуетесь для работы с TeX-ом?

- Блокнот и командная строка;
- WinEdt (а Вы знали, что это платная программа?);
- Eclipse и TeXlipse  
(<http://texlipse.sourceforge.net/release.html>);
- Kile (только для линуксойдов :));

# Eclipse

## Основные фиши TeXlipse

- Контекстная подсказка основных команд (меню с основными командами);



# Eclipse

## Основные фишки TeXlipse

```
\usepackage[english,russian]{babel}
\usepackage{amssymb}
\usepackage{pgf}
\usepackage{tikz}
\usepackage{multicol}
\usepackage{url}

\title{TeX: первые шаги к комфорту}

\author{Алексей Хворост}

\date{Теоретические компьютерные науки}
```

```
\new
```

```
\beg
```

```
\beg
```

```
\tit
```

```
\end
```

```
\sec
```

```
\sec
```

```
\beg
```

```
\fra
```

```
\beg
```

```
\i
```

```
\end{itemize}
```

```
\end{frame}
```

newcommand  
newenvironment  
newline  
newpage

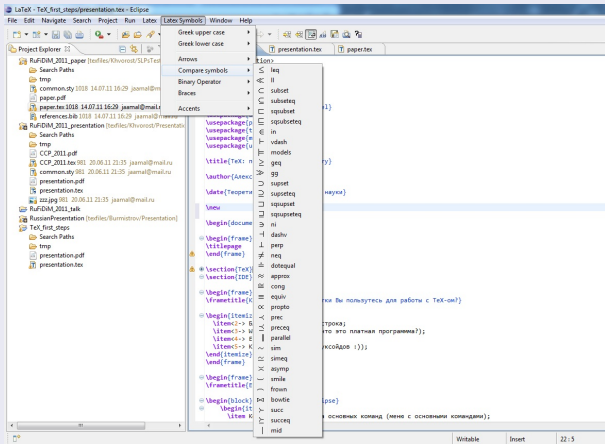
### \newcommand

Defines a new command. An optional argument between the two mandatory ones defines the number of arguments of the command. Arguments can be referred in the definition as #1, #2 etc.

```
\item<4-> Eclipse и TeXlipse;
\item<5-> Kile (только для линуксойдов :));
\end{itemize}
```

# Eclipse

## Основные фиши TeXlipse



# Eclipse

## Основные фиши TeXlipse

- Контекстная подсказка основных команд (меню с основными командами);
- Контекстная подсказка для пользовательских стилевых пакетов;

# Eclipse

## Основные фиши TeXlipse

It is well-known that adding a new rule to an AVL-tree is quite a complex operation. Every adding operation generates a sequence of rotations of the tree. There are two types of rotations symbolically presented in Figure 1, see `\cite{SLPConstruction}` for more details. Every rotation may generate at most three new rules. Also every rotation may generate three unused rules. There are two possible directions for an optimization of the algorithm: to construct more compact grammar and to minimize the number of queries to AVL-trees. Minimizing of the number of queries to AVL-trees becomes important when the size of input text becomes huge and we cannot store an AVL-tree in the memory. Formally it means that costs of a query to an AVL-tree are greater than costs of calculations in memory.

`\cite{B}`

The fol

SLPConstruction  
SmallestCFG

article  
author: W. Rytter  
title: Application of Lempel-Ziv factorization to the  
approximation of grammar-based compression  
journal: Theor. Comput. Sci.  
volume: 302  
number: 1-3  
year: 2003  
pages: 211-222

`\noindent`  
built a  
LZ-fact

Let us

`$$$((\backslash s1`

upper b

`\begin{`

`\sum_{i=1}^{n-1}`

`\end{mu`

Notice

algorit

reverse order  $\backslash s1p[5] \cdot \backslash s1p[F]_1 \cdot \dots \cdot \backslash s1p[F]_{n-2} \cdot \backslash s1p[F]_{n-1}$ , we get the following estimation:  
$$$$$ \sum_{i=1}^{n-1} ( \log(2^{n-i}) - \log(2^{n-i-1}) ) = (n-1) \log(2) = \Theta(n). $$$$$

Our next example shows that several factors can be processed together if they occur in a single SLP.

`\noindent \textbf{Example 2:}` Let  $n > 0$  be an integer and  $S = b \cdot a^{2^{n-1}} \cdot b \cdot a^{2^{n-2}} \cdot \dots \cdot b \cdot a$ . So the length of  $S$  is equal to  $2^n + n - 2$ . Consider the LZ-factorization of  $S$ :  
$$b \cdot a \cdot a^2 \cdot a^4 \cdot \dots \cdot a^{2^{n-2}-1} \cdot b \cdot a^{2^{n-2}} \cdot a^{2^{n-3}} \cdot \dots \cdot b \cdot a.$$
 Let  $S1p[5]_1$  be an SLP that derives  $b \cdot a^{2^{n-1}}$ . It is obvious that all other factors starting with  $b \cdot a^{2^{n-2}}$  occur in  $S1p[5]_1$ , therefore the algorithm may process them together. So

# Eclipse

## Основные фиши TeXlipse

- Контекстная подсказка основных команд (меню с основными командами);
- Контекстная подсказка для пользовательских стилевых пакетов;
- Контекстная подсказка ошибок и предупреждений;

# Eclipse

## Основные фиши TeXlipse

```
@\section{Introduction}
```

Nowadays searching algorithms on huge data sets attract much attention. To reduce the input size one needs algorithms that can work directly with a compressed representation of input data.

❗ [Undefined control sequence: '\unknowncommand' (followed by:)]

Various compressed representations of strings are known: straight-line programs (SLPs)~\cite{SLPConstruction}, collage-systems \cite{collages}, string representations using antidictionaries \cite{antidictionaries}, etc. Nowadays text compression based on context-free grammars such as SLPs has become a popular research direction. The reason for this is not only that grammars provide well-structured compression but also that the SLP-based compression is, in a sense, polynomially equivalent to the compression achieved by the Lempel-Ziv algorithm that is widely used in practice. It means that, given a text \$\$\$, there is a polynomial relation between the size of an SLP that derives \$\$\$ and the size of the dictionary stored by the Lempel-Ziv algorithm, see \cite{SLPConstruction}. It should also be noted that classical LZ78 \cite{LZ78} and LZW \cite{LZW} algorithms can be considered as special cases of grammar compression. (At the same time other compression algorithms from the Lempel-Ziv family---such as LZ77 \cite{LZ77} and run-length compression---do not fit directly into grammar compression model.)

Using the fact that SLPs are nicely structured, several researchers keep developing analogues of classical string algorithms that (at least theoretically) perform quite well on SLP-compressed representations: \textbf{Pattern matching} \cite{PM\_and\_MD}, \textbf{Longest common substring} \cite{LCS substring}, \textbf{Computing all palindromes} \cite{LCS substring}, some versions of \textbf{Longest common subsequence} \cite{LCS\_P}. At the same time, constants hidden in big-O notation for algorithms on SLPs are often very big. Also the aforementioned polynomial relation between the size of an SLP for a given text and the size of the LZ77-dictionary for the same text does not yet guarantee that SLPs provide good compression ratio in practice. Thus, a major questions is whether or not there exist SLP-based compression models suitable to practical usage? This question splits into two sub-questions addressed in the present paper: How difficult is it to compress data to an SLP-representation? How large compression ratio do SLPs provide as compared to classic algorithms used in practice?

# Eclipse

## Основные фиши TeXlipse

- Контекстная подсказка основных команд (меню с основными командами);
- Контекстная подсказка для пользовательских стилевых пакетов;
- Контекстная подсказка ошибок и предупреждений;
- Проверка орфографии (английский по-умолчанию, для русского надо скачать словарь);

# Eclipse

## Основные фишки TeXlipse

`\section{Introduction}`

Nowadays searching algorithms on huge data sets attract much attention. To reduce the input size one needs algorithms that can work directly with a compressed representation of input data.

Various compressed representations of strings are known: straight-line programs (SLPs)~\cite{SLPConstruction},

collage-systems~\cite{antidictionaries}, etc. Nowadays text compression is not only a popular research direction. The reason for this is not only that the SLP-based compression is, in a sense, polynomial, but also that the SLP-based compression is, in a Lempel-Ziv algorithm that is widely used in practice. It means that, between the size of an SLP that derives \$\$\$ and the size of the dictionary, there is a polynomial relation. It should also be noted that classical LZ76 \cite{LZ76} is special cases of grammar compression. (At the same time other compressed representations of strings, such as LZ77 \cite{LZ77} and run-length compression---do not fit directly into the SLP framework.)

Using the fact that SLPs keep developing analogues of classical string algorithms that work with compressed representations: \textbf{Pattern matching}~\cite{PM\_and\_HD}, \textbf{Computing all palindromes}~\cite{LCSsubstring}, some versions of \textbf{Pattern matching}~\cite{LCSsubstring}, some the same time, constants hidden in big-O notation for polynomial relation between the size of an SLP for a given text and the size of the dictionary. SLPs provide good compression ratio in practice. Thus, a major question is whether or not there exist SLP-based compression models suitable for practical usage? This question splits into two sub-questions addressed in the present paper: How difficult is it to compress data to an SLP-representation? How large compression ratio do SLPs provide as compared to classic algorithms used in practice?

Let us describe in more detail the content of the paper and its structure. Section 2 gathers some preliminaries about SLPs. In Section 3 we present an improved version of Rytter's algorithm~\cite{SLPConstruction} for constructing an SLP-representation of a given text. In Section 4 we compare the improved algorithm vs. the original algorithm from~\cite{SLPConstruction} and also present results of a comparison of compression ratio between the two SLP-based

- Change to "compressed"
- Change to "compressed"
- Ignore "compressed" during the current session
- Add "compressed" to dictionary



# Eclipse

## Основные фишки TeXlipse

- Контекстная подсказка основных команд (меню с основными командами);
- Контекстная подсказка для пользовательских стилевых пакетов;
- Контекстная подсказка ошибок и предупреждений;
- Проверка орфографии (английский по-умолчанию, для русского надо скачать словарь);
- Автоматическая компиляция при сохранении файла;
- Абсолютно бесплатный софт;
- Обновления он-лайн;
- Интеграция с системами контроля версий;

# Eclipse

## Основные фиши TeXlipse

- Контекстная подсказка основных команд (меню с основными командами);
- Контекстная подсказка для пользовательских стилевых пакетов;
- Контекстная подсказка ошибок и предупреждений;
- Проверка орфографии (английский по-умолчанию, для русского надо скачать словарь);
- Автоматическая компиляция при сохранении файла;
- Абсолютно бесплатный софт;
- Обновления он-лайн;
- Интеграция с системами контроля версий;

# Eclipse

## Основные фиши TeXlipse

- Контекстная подсказка основных команд (меню с основными командами);
- Контекстная подсказка для пользовательских стилевых пакетов;
- Контекстная подсказка ошибок и предупреждений;
- Проверка орфографии (английский по-умолчанию, для русского надо скачать словарь);
- Автоматическая компиляция при сохранении файла;
- Абсолютно бесплатный софт;
- Обновления он-лайн;
- Интеграция с системами контроля версий;

# Eclipse

## Основные фиши TeXlipse

- Контекстная подсказка основных команд (меню с основными командами);
- Контекстная подсказка для пользовательских стилевых пакетов;
- Контекстная подсказка ошибок и предупреждений;
- Проверка орфографии (английский по-умолчанию, для русского надо скачать словарь);
- Автоматическая компиляция при сохранении файла;
- Абсолютно бесплатный софт;
- Обновления он-лайн;
- Интеграция с системами контроля версий;

# Презентации

- Вы часто делаете презентации?
- Ваш ритм работы: внес изменения → компиляция → закрыть Adobe Reader → открыть презентацию;
- Странное ощущение: почему приходится тратить время на закрытие Adobe Reader и его повторное открытие?

# Презентации

- Вы часто делаете презентации?
- Ваш ритм работы: внес изменения → компиляция → закрыть Adobe Reader → открыть презентацию;
- Странное ощущение: почему приходится тратить время на закрытие Adobe Reader и его повторное открытие?

# Презентации

- Вы часто делаете презентации?
- Ваш ритм работы: внес изменения → компиляция → закрыть Adobe Reader → открыть презентацию;
- Странное ощущение: почему приходится тратить время на закрытие Adobe Reader и его повторное открытие?

# Okular

Okular (<http://windows.kde.org/>)

- Автоматическое обновление файла экономит массу времени;
- Абсолютно бесплатный софт;
- Инсталлировать чуть сложнее чем Adobe Reader;



# Okular

## Okular (<http://windows.kde.org/>)

- Автоматическое обновление файла экономит массу времени;
- Абсолютно бесплатный софт;
- Инсталлировать чуть сложнее чем Adobe Reader;

# Okular

## Okular (<http://windows.kde.org/>)

- Автоматическое обновление файла экономит массу времени;
- Абсолютно бесплатный софт;
- Инсталлировать чуть сложнее чем Adobe Reader;

# Okular

## Okular (<http://windows.kde.org/>)

- Автоматическое обновление файла экономит массу времени;
- Абсолютно бесплатный софт;
- Инсталлировать чуть сложнее чем Adobe Reader;

# Презентации

- Вы тратите львиную долю время на верстку картинок?
- Вам надоело прорисовывать каждую точку руками?
- В поисковиках столько картинок, неужели их так сложно рисовать?

# Презентации

- Вы тратите львиную долю время на верстку картинок?
- Вам надоело прорисовывать каждую точку руками?
- В поисковиках столько картинок, неужели их так сложно рисовать?

# Презентации

- Вы тратите львиную долю время на верстку картинок?
- Вам надоело прорисовывать каждую точку руками?
- В поисковиках столько картинок, неужели их так сложно рисовать?

# Презентации

- Вы тратите львиную долю время на верстку картинок?
- Вам надоело прорисовывать каждую точку руками?
- В поисковиках столько картинок, неужели их так сложно рисовать?

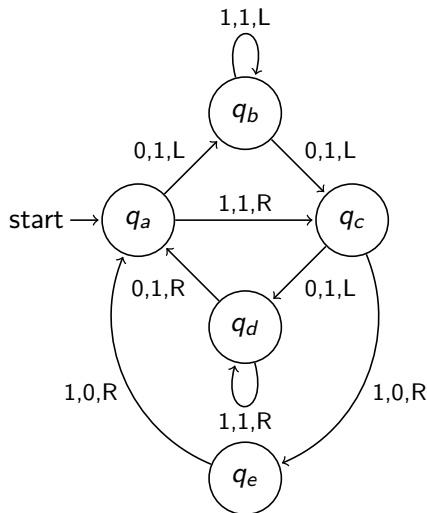
# Till Tantau



- beamer
- TikZ



Сколько необходимо времени чтобы нарисовать этот автомат?



# Код, который получается с помощью TikZ

```

\begin{center}
\begin{tikzpicture}[->,shorten >=1pt,auto,node distance=2cm,on grid,semithick,inner sep=2pt,bend angle=45,scale=0.8]
  \node[initial,state] (A)      {$q_a$};
  \node[state]          (B) [above right=of A] {$q_b$};
  \node[state]          (D) [below right=of A] {$q_d$};
  \node[state]          (C) [below right=of B] {$q_c$};
  \node[state]          (E) [below=of D]      {$q_e$};

  \path [every node/.style={font=\footnotesize}]
    (A) edge              node {0,1,L} (B)
        edge              node {1,1,R} (C)
    (B) edge [loop above] node {1,1,L} (B)
        edge              node {0,1,L} (C)
    (C) edge              node {0,1,L} (D)
        edge [bend left]  node {1,0,R} (E)
    (D) edge [loop below] node {1,1,R} (D)
        edge              node {0,1,R} (A)
    (E) edge [bend left]  node {1,0,R} (A);
\end{tikzpicture}
\end{center}

```

Вы уже все бросили и побежали ставить обновления? :)

