

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Московский государственный университет имени М. В. Ломоносова»



ФИЗИЧЕСКИЙ ФАКУЛЬТЕТ

Учебно-методическое пособие

**ПОСОБИЕ ДЛЯ СТУДЕНТОВ ПО ОФОРМЛЕНИЮ
КУРСОВЫХ, ДИПЛОМНЫХ И ДРУГИХ РАБОТ В СИСТЕМЕ
ВЁРСТКИ НАУЧНЫХ ТЕКСТОВ L^AT_EX**

Выполнил:

к. ф.-м. н. доц. кафедры мате-
матического моделирования и
информатики А. В. Зубюк

Москва – 2020

СОДЕРЖАНИЕ

Введение	3
Глава 1 Система вёрстки научных текстов \LaTeX и её использование	4
1.1 Что такое \LaTeX и зачем он нужен	4
1.2 Структура исходных текстов на языке \LaTeX	4
1.3 Подготовка (редактирование) исходных текстов на языке \LaTeX	5
1.4 Сборка (компиляция) PDF-файла из исходных текстов на языке \LaTeX	5
1.5 Сборка с сохранением временных файлов в отдельной директории	7
1.6 Сборка с использованием утилиты <code>make</code>	8
1.7 Сборка в редакторе <code>TeXMaker</code>	8
Глава 2 Оформление исходных текстов на языке \LaTeX	10
2.1 Набор обычного текста, пробелы, пунктуация	10
2.2 Разделы, формулы, их нумерация и ссылки на них	12
2.3 Оформление рисунков и таблиц	15
2.4 Ускорение набора исходных текстов с помощью определения собственных команд (макросов)	16
2.5 Набор теорем, лемм, следствий, доказательств, определений, примеров и замечаний	17
2.6 Оформление списка литературы и ссылок на литературу	18
Заключение	19
Список использованных источников	20
Приложение А Зачем нужны приложения.	22
Приложение Б Справочник часто используемых специальных символов и команд \LaTeX	23

ВВЕДЕНИЕ

Как и многие другие научные руководители, я — автор этот пособия — в какой-то момент осознал, что проще один раз написать краткий «курс молодого бойца» по оформлению курсовых и дипломных студенческих работ, чем проводить такой инструктаж персонально для каждого студента, работающего под моим руководством. Результат моих усилий — перед вами.

Впрочем, как показывает мой опыт, он может быть полезен и опытным научным работникам, давно работающим с \LaTeX .

В настоящем пособии на примерах показано, как следует оформлять основные элементы курсовой или дипломной работы (титульный лист, разделы, формулы, рисунки, таблицы и др.) в системе вёрстки научных текстов \LaTeX . О самой системе \LaTeX рассказано в главе 1. Основные правила оформления текстов с примерами даются, начиная с главы 2.

Техническая информация о работе с системой \LaTeX в настоящем пособии перемежается общими рекомендациями по написанию научных текстов, информацией о том, как должна быть структурирована работа, какая информация должна содержаться в тех или иных её разделах.

Студентам предлагается использовать исходный текст настоящего пособия как шаблон своей работы. Его можно скачать с сайта научной группы, руководимой автором пособия, по ссылке

<http://neurofuzzy.phys.msu.ru/~zubuk/DiplomaTemplate.zip>.

Начать изучение предлагается с преобразования скачанного исходного текста в PDF-файл согласно инструкциям, данным в разделах 1.4–1.7. После этого каждый студент, прочитавший настоящее пособие и изучивший его исходный текст, сможет легко преобразовать его под свои нужды. Помогут в этом имеющиеся в исходном тексте пособия комментарии.

Оформление пособия основано на разработанном его автором стилевом файле `NeuroFuzzy.sty`, который подключается в начале документа:

```
1 \usepackage{NeuroFuzzy}
```

Оно в основном (но не полностью) соответствует требованиям [ГОСТ 7.32-2017 «Отчёт о научно-исследовательской работе: структура и правила оформления»](#).

Глава 1 Система вёрстки научных текстов \LaTeX и её использование

1.1 Что такое \LaTeX и зачем он нужен

\LaTeX — это, пожалуй, самая популярная в мире система вёрстки научных текстов. Все сколь-нибудь значимые научные журналы в области физики и математики оформляются в \LaTeX (это «Nature», «Fuzzy Sets and Systems», «Вестник Московского университета» и многие многие другие), его используют ведущие зарубежные и отечественные издательства научной литературы (например, Elsevier и ФИЗМАТЛИТ).

Именно поэтому в нашей научной группе для подготовки курсовых, дипломных и др. работ мы настоятельно рекомендуем студентам использовать \LaTeX .

Также \LaTeX удобно использовать для подготовки презентаций. Эта возможность особенно полюбилась математикам из-за удобства вставки формул, поэтому подавляющее большинство презентаций на международных математических конференциях подготовлено в \LaTeX .

Система вёрстки научных текстов \LaTeX представляет собой совокупность двух компонент:

- 1) *Специальный язык разметки*. Наверняка вы слышали о языке разметки HTML, используемом для оформления web-страниц. Язык \LaTeX служит аналогичной цели, но предназначен для оформления научных текстов;
- 2) *Компилятор*, который преобразует исходный текст на языке \LaTeX (файл с расширением `.tex`), в PDF-файл.

Дистрибутивы компилятора \LaTeX можно найти для всех основных операционных систем: для Windows можно использовать дистрибутив MikTeX, для Linux и др. Unix-подобных систем — TeXlive.

1.2 Структура исходных текстов на языке \LaTeX

Как правило, исходный текст документа состоит из нескольких файлов:

- *Файл с расширением .tex* — основной файл документа, в котором набран его текст;
- *Файл с расширением .bib* — файл, в котором в специальном формате BibTeX хранится информация об источниках, на которые в документе даются ссылки (библиография). Сохранять информацию в формате BibTeX умеют все основные библиографические базы: [Google Scholar](#), [Web of Science](#), [SCOPUS](#) и др.;
- *Дополнительные файлы*, например, изображения в форматах PDF, JPEG, PNG и др. или данные для построения графиков. В настоящем шаблоне они хранятся в директории `pictures`.

1.3 Подготовка (редактирование) исходных текстов на языке L^AT_EX

Исходные тексты на языке L^AT_EX — это файлы с обычным текстом, их можно создавать и редактировать в любом текстовом редакторе. В Windows это может быть редактор «Блокнот» («Notepad»), в Linux и др. Unix-подобных операционных системах — редакторы «GEdit», «Kate» и др.

Однако, наиболее удобно это делать в специальных редакторах, которые умеют «подсвечивать» синтаксические конструкции L^AT_EX, а также сразу преобразовывать исходные тексты документа в PDF-файл с помощью компилятора L^AT_EX. Одним из наиболее популярных редакторов сегодня является T_EXMaker.

1.4 Сборка (компиляция) PDF-файла из исходных текстов на языке L^AT_EX

В программировании *сборкой* (или *компиляцией*) называют преобразование исходных текстов программы (например, написанных на языке C++) в исполняемый программный модуль (например, в файл с расширением `.exe`, если речь идёт об операционной системе Windows). В случае работы с L^AT_EX сборкой называют преобразование исходных текстов на языке L^AT_EX в итоговый PDF-файл. Она осуществляется в несколько этапов:

- 1) Первый проход утилиты-компилятора `pdflatex`:

```
1 pdflatex -shell-escape diploma.tex
```

На этом этапе формируется PDF-файл, однако, этот файл не является окончательным: в нём отсутствуют разделы «Содержание» (оглавление) и «Список использованных источников», могут отсутствовать ссылки на некоторые нумерованные объекты (разделы, формулы, рисунки и др.).

При этом создаются *временные*¹ файлы, в которых сохраняется информация о том, какие номера присвоены разделам, формулам, рисункам и др. нумерованным объектам документа, о том, на каких страницах они расположены, о том на какие библиографические источники имеются ссылки в документе и т. д.

Невозможность сразу сформировать окончательный PDF-файл связана с особенностью работы всех компиляторов (в т. ч. компиляторов программ на языках C/C++ и др.) — при анализе файла с исходными текстами они движутся сверху вниз и никогда не возвращаются обратно. Поэтому одного прохода утилиты-компилятора `pdflatex` недостаточно для формирования оглавления (т. к. неизвестно, на каких страницах окажутся разделы документа), правильных ссылок на нумерованные объекты (т. к. ссылка может встретиться в исходном тексте раньше, чем сам нумерованный объект, т. е. до того, как ему будет присвоен номер) и т. д.

2) Формирование библиографии утилитой `biber`:

```
1 biber diploma
```

На этом этапе на основе собранной ранее информации об использованных библиографических источниках в специальном временном файле формируется исходный текст раздела «Список использованных источников».

3) Второй проход утилиты `pdflatex`:

```
1 pdflatex -shell-escape diploma.tex
```

¹Несмотря на эпитет «временные» эти файлы не удаляются автоматически и остаются на диске после завершения сборки.

На этом этапе формируется более полный, но всё же ещё не окончательный PDF-файл, в который из сформированных ранее временных файлов вставляется содержимое разделов «Содержание», «Список использованных источников», на места ссылок на нумерованные объекты подставляются номера этих объектов и т. д. Всей этой информации не было в PDF-файле, сформированном при первом проходе утилиты `pdflatex`. Однако, из-за вставки нового материала некоторые разделы могут оказаться на других страницах в сравнении с PDF-файлом, сформированным при первом проходе утилиты `pdflatex`. В связи с этим номера страниц в разделе «Содержание» необходимо обновить, для чего служит следующий шаг.

4) Третий проход утилиты `pdflatex`:

```
1 pdflatex -shell-escape diploma.tex
```

На этом этапе формируется итоговый PDF-файл.

1.5 Сборка с сохранением временных файлов в отдельной директории

При использовании описанного в предыдущем разделе способа сборки PDF-файла директория с исходными текстами «замусоривается» большим количеством временных файлов. Удаление этих файлов после окончания работы с документом — достаточно неприятный и трудоёмкий процесс. В связи с этим разумно создать отдельную директорию для хранения временных файлов, например, директорию `build`, и с помощью флага `--output-dir` «заставить» используемые утилиты сохранять все временные файлы в ней. В этом случае сборка PDF-файла может быть осуществлена следующим образом:

```
1 mkdir build
2 pdflatex --output-dir=build -shell-escape diploma.tex
3 biber --output-dir=build diploma
4 pdflatex --output-dir=build -shell-escape diploma.tex
5 pdflatex --output-dir=build -shell-escape diploma.tex
```

Замечание 1. При сборке исходных текстов настоящего документа с сохранением временных файлов в директории `build` необходимо в на-

чале файла `diploma.tex` при подключении пакета `NeuroFuzzy` установить опцию `usebuilddir`:

```
1 \usepackage[usebuilddir]{NeuroFuzzy}
```

В противном случае сборка приведёт к ошибкам. Это замечание верно и при сборке с использованием редактора `TeXMaker`, описанной в разделе 1.7.

1.6 Сборка с использованием утилиты `make`

Ещё более удобным способом сборки является использование утилиты `make` и заранее подготовленного файла `Makefile`, в котором прописаны инструкции по сборке, используемые утилитой `make`. При таком подходе достаточно просто зайти в директорию, где хранятся `Makefile` и основной файл документа, и выполнить команду

```
1 make
```

Впрочем, такой способ доступен только пользователям `Linux` и др. Unix-подобных операционных систем

1.7 Сборка в редакторе `TeXMaker`

Сборку документа удобно осуществлять параллельно с его редактированием в редакторе `TeXMaker`. Для этого нужно зайти в редактор и настроить параметры сборки, см. рисунок 1. После этого можно выполнять действия, описанные в разделах 1.4 и 1.5, выбирая соответствующие пункты меню «Tools»:

- для одного прохода утилиты `pdflatex` следует выбрать пункт «Tools → PDFL^AT_EX», «горячая» клавиша `F6`;
- для одного прохода утилиты `biber` следует выбрать пункт «Tools → BibT_EX», «горячая» клавиша `F11`.

Таким образом, полная сборка результирующего PDF-файла может быть осуществлена следующей последовательность «горячих» клавиш:

`F6 → F11 → F6 → F6`.

Для просмотра результата используйте пункт меню «Tools → View PDF», «горячая» клавиша `F7`. Также удобно использовать «быструю» сборку «Tools → Quick Build», «горячая» клавиша `F1`.

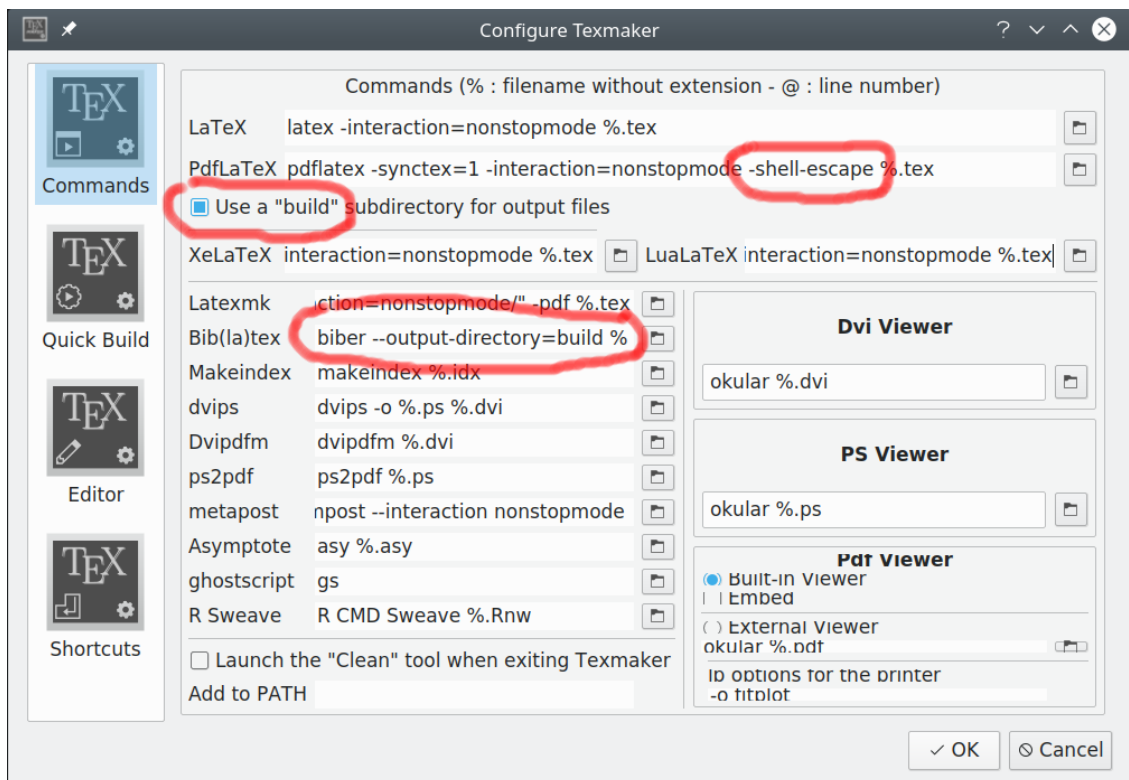


Рисунок 1 — Настройка редактора \TeX Maker (пункт меню «Options → Configure Texmaker») для правильной сборки настоящего документа. Обратите внимание на флаг `-shell-escape` и использование утилиты `biber` вместо `bibtex`. Для удобства \TeX Maker настраивается на использование поддиректории `build` для хранения временных файлов, соответственно, обратите внимание на замечание 1, сделанное выше в разделе 1.5.

Глава 2 Оформление исходных текстов на языке \LaTeX

2.1 Набор обычного текста, пробелы, пунктуация

Исходный текст \LaTeX в основном файле документа с расширением `.tex` набирается обычным способом, как это делается в любом текстовом редакторе.

Слова в исходном тексте разделяются пробелами (любым количеством) или переносом строки, при сборке PDF-файла эти символы заменяются на *один* пробел. Таким образом, если при наборе текста вы случайно вставите между словами два или более пробелов, при компиляции «лишние» пробелы не будут учтены. Естественно, что использовать множественные пробелы для выравнивания текста в \LaTeX бессмысленно — все они заменятся на одинарный пробел.

Абзацы в исходном тексте отделяются одной или несколькими пустыми строками:

- 1 Текст одного абзаца.
- 2
- 3 Текст другого абзаца.

Красные строки (отступы в начале абзаца) \LaTeX проставляет автоматически.

При наборе исходного текста следует обратить внимание на специальные пробелы и знаки пунктуации, используемые в \LaTeX , см. таблицу 1. Без их правильного использования итоговый текст в PDF-файле будет выглядеть небрежно.

Неразрывные пробелы следует вставлять между словами, которые не должны разделяться переносом строки или переходом на другую страницу. Например, между словом «таблица» и номером таблицы 1 в ссылке на неё. При этом различают неразрывные пробелы переменной и фиксированной длины. Первые имеют ту же длину, которую имеют все пробелы в данной строке (длины пробелов в разных строках различаются, за счёт этого текст выравнивается по ширине страницы). Вторые имеют фиксированную длину. Типовые примеры применения неразрывных пробелов даны в таблице 1.

Таблица 1 — Специальные пробелы и знаки пунктуации в L^AT_EX.

Описание специального пробела или знака препинания	Пример применения	Специальный символ или команда L ^A T _E X
Неразрывный пробел переменной длины	таблица 1	~
Короткий неразрывный пробел	А. В. Зубюк (между А. и В.)	\,
Длинный неразрывный пробел	А. В. Зубюк (между инициалами и фамилией)	\;
Дефис (короткая черта)	физ.-мат. науки	-
Минус (средняя черта)	стр. 1–10	--
Тире (длинная черта)	Тире — длинная черта (между знаком тире и предшествующим словом рекомендуется ставить неразрывный пробел ~, чтобы тире не переносилось на другую строку или страницу)	---
Русские кавычки-«ёлочки»	Метро «Университет»	<< (откр.) >> (закр.)
Английские кавычки-«лапки»	Next stop is “University”	` ` и ` ` (см. буквы «ё» и «э» на клавиатуре)

К специальным знакам пунктуации отнесём разные виды черт, применяющиеся в разных синтаксических конструкциях: дефис, минус и тире. Также к ним отнесём кавычки. Для их набора в \LaTeX используются специальные сочетания символов. Типовые примеры их применения даны в таблице 1.

2.2 Разделы, формулы, их нумерация и ссылки на них

Заголовки разделов в \LaTeX создаются командами (макросами) `\part`, `\chapter`, `\section`, `\subsection` и т.д. Все разделы нумеруются автоматически. Для того, чтобы сослаться на раздел, после команды, формирующей его заголовок, необходимо вставить *метку* с помощью команды `\label{метка}`, где *метка* — придуманный вами уникальный идентификатор раздела, по которому на него можно будет ссылаться. Например, настоящий раздел 2.2 на стр. 12 под названием «Разделы, формулы, их нумерация и ссылки на них» в исходном тексте начинается так:

```
1 \section{Разделы, формулы, их нумерация и ссылки на них}
2 \label{sec:sections}
```

Для ссылок на него использованы команды `\ref{sec:sections}` (номер раздела), `\pageref{sec:sections}` (страница, где начинается раздел) и `\nameref{sec:sections}` (текст заголовка).

Аналогично (с помощью команд `\label`, `\ref`, `\pageref`) оформляются ссылки на любые другие нумерованные объекты: таблицы, рисунки и т.п. Ссылки на формулы создаются с помощью команды `\eqref`, которая включает номер формулы в круглые скобки, как это принято в научной литературе.

Формулы бывают *обособленными* (на отдельной строке) и *встроенными* (внутри обычного текста). Обособленные формулы могут нумероваться, а могут — нет. Для создания встроенных формул используются знаки доллара: $\$формула\$$. Для создания нумерованных формул следует использовать окружения `equation`, `multline`, `gather`, `align` и др., например:

```
1 \begin{equation}
2     формула
3 \end{equation}
```

Нумеровать следует только те формулы, на которые в тексте есть ссылки.

Например, формула (1) позволяет связать площадь круга S с его радиусом r :

$$S = \pi r^2. \quad (1)$$

А вот выражение для длины окружности l :

$$l = 2\pi r.$$

На него в тексте нет ссылки, поэтому использованы скобки `\[` и `\]` для создания нумерованной обособленной формулы. Также нумерованные формулы можно создавать с помощью окружений «со звёздочкой»: `equation*`, `multline*`, `gather*`, `align*` и др.

Внутри формул вы можете использовать всевозможные специальные знаки: индексы, греческие буквы, интегралы и т. д. Для их набора следует использовать специальные команды и символы L^AT_EX:

— нижние и верхние индексы: `ai`, `aj`, `Aij`

$$a^i, a_j, A_j^i;$$

— греческие буквы: `\alpha`, `\beta`, `\gamma`, `\Gamma`, `\omega`, `\Omega` и др.

$$\alpha, \beta, \gamma, \Gamma, \omega, \Omega;$$

— интегралы: `\int\limits_0^\infty`, `\iint`, `\oint` и др.

$$\int_0^\infty, \iint, \oint;$$

— сумма `\sum_{i=1}^n` и произведение `\prod_{i=1}^n`

$$\sum_{i=1}^n, \prod_{i=1}^n;$$

— дроби: `\frac{a}{b}`, `\dfrac{a}{b}`

$$\frac{a}{b}, \frac{a}{b};$$

- символы математических функций, операторов и отношений: `\sin`, `\exp`, `\partial`, `\nabla`, `\sim`, `\approx`, `\in`, `\subset`, `\notin`, `\min_a`, `\max_a`, `\inf_a`, `\sup_a`, `\lim_{a\to b}` и др.

$$\sin, \exp, \partial, \nabla, \sim, \approx, \in, \subset, \notin, \min_a, \max_a, \inf_a, \sup_a, \lim_{a \rightarrow b};$$

- векторы и другое декорирование: `\vec a`, `\tilde a`, `\hat a`, `\widetilde{abc}` и др.

$$\vec{a}, \tilde{a}, \hat{a}, \widetilde{abc};$$

- матрицы:

```
1 \begin{pmatrix}1 & 2 & 3\\ 4 & 5 & 6\end{pmatrix}
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix};$$

- стрелки: `\xrightarrow[n\to\infty]{\text{п.н.}}`, `\to`, `\rightarrow`, `\Leftarrow`

$$\xrightarrow[n \rightarrow \infty]{\text{п. н.}}, \rightarrow, \Rightarrow, \Leftarrow;$$

- многострочные формулы, системы уравнений, скобки

```
1 \begin{gather*}
2 S = \pi r^2, l = 2\pi r, \\
3 |a| = \begin{cases} a, & a > 0, \\ -a, & a \leq 0, \end{cases} \\
4 \left\{ \int\limits_{-\infty}^{\infty} f(x, c) dx \;\middle|\; c \in \right. \\
5 \quad \left. \leftarrow [0, 1] \right\}.
```

$$S = \pi r^2, \quad l = 2\pi r,$$

$$|a| = \begin{cases} a, & a > 0, \\ -a, & a \leq 0, \end{cases}$$

$$\left\{ \int\limits_{-\infty}^{\infty} f(x, c) dx \;\middle|\; c \in [0, 1] \right\}.$$

2.3 Оформление рисунков и таблиц

Рисунки и таблицы принято считать «плавающими» объектами. Это означает, что они могут располагаться в тексте не там, где даётся их описание. Так, например, согласно ГОСТ рисунок может располагаться на той странице, где даётся его описание (точнее — ссылка на него), или же на следующей странице. Иногда все рисунки и таблицы печатают к концу документа.

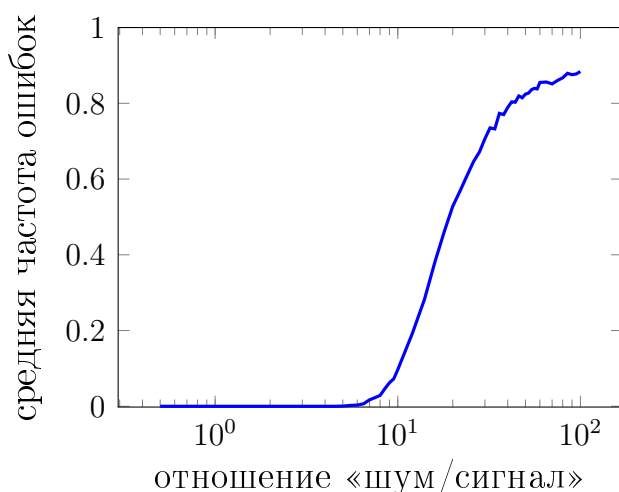
L^AT_EX автоматически выбирает наиболее подходящее расположение для рисунков и таблиц. Без крайней необходимости не стоит вмешиваться в этот механизм и пытаться «заставить» рисунок или таблицу расположиться в каком-то ином месте.

Важно помнить, что каждый рисунок и каждая таблица должны быть подписаны, т. е. должны иметь заголовки. Заголовки создаются с помощью команды `\caption`, которая сама печатает постоянную часть заголовка (слово «Рисунок» или «Таблица»), проставляет очередной номер, печатает символ-разделитель (тире). Также важно помнить, что на каждый рисунок и каждую таблицу в основном тексте документа *обязательно* должна быть ссылка, сделанная с помощью команды `\label`.

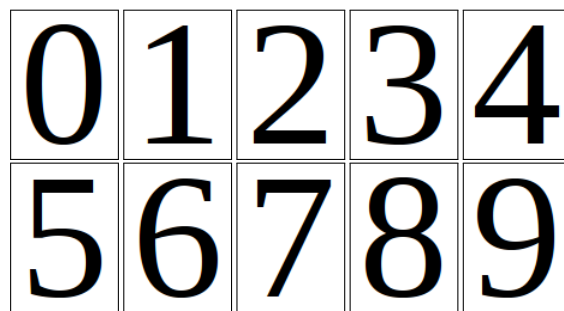
Пример таблицы был дан выше, см. таблицу 1, как и пример обычного рисунка (т. е. рисунка, который состоит из изображения, вставленного из графического файла), см. рисунок 1.

В общем-то, любой рисунок можно оформить простой вставкой изображения из графического файла. Однако, в научных работах мы очень часто имеем дело с графиками, построенными по данным физического эксперимента, расчётов и т. п. Для построения таких графиков лучше использовать средства самого L^AT_EX, т. к. это позволит оформить все графики в работе в едином стиле (одинаковые толщины линий, шрифты подписей и т. д.), согласованном со стилем основного текста (размеры шрифтов, начертание символов, в т. ч. греческих букв, и т. п.).

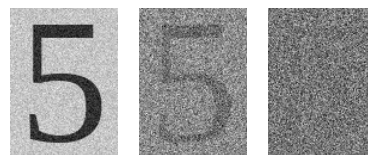
При использовании «внешних» программ для подготовки графиков соблюсти единообразие оформления не удастся, и графики будут выглядеть небрежно. Кроме того, графики, оформленные средствами L^AT_EX, легко переносятся из исходных текстов, предназначенных для печати на бумаге в формате A4, в презентации (конечно же, речь идёт о презентациях,



(а) Зависимость средней частоты ошибок идентификации от отношения «шум/сигнал» корня из дисперсии аддитивного шума, искажающего предъявленные изображения, к разности максимальной и минимальной яркостей неискажённых изображений.



(б) Изображения цифр «0»–«9», не искажённые шумом.



(в) Изображения цифры «5», демонстрирующие интенсивность шума при отношении «шум/сигнал», равном 0.1, 1 и 10.

Рисунок 2 — Иллюстрация работы морфологического алгоритма идентификации печатных цифр «0»–«9», искажённых случайным аддитивным шумом.

сделанных в \LaTeX), т. к. при сборке PDF-файла презентации оформление графиков автоматически будет приведено к стилю презентации. При использовании «внешних» программ графики для презентаций придётся делать заново, т. к. подписи графиков в презентациях принято делать *значительно* крупнее, чем в печатных текстах (иначе их просто не будет видно слушателям).

Для построения графиков средствами \LaTeX данные, по которым строятся графики, следует сохранить в форме таблицы в обычный текстовый файл и использовать средства пакета `tikz` и окружения `\begin{tikzpicture}... \end{tikzpicture}`, см. рисунок 2(а), построенный по данным из файла `pictures/identification.dat`. На рисунке 3 показано, как строить «столбиковые» гистограммы и делать легенды на графиках.

2.4 Ускорение набора исходных текстов с помощью определения собственных команд (макросов)

Набор исходных текстов можно *кардинально ускорить*, если для часто повторяющихся фрагментов (например, сложных математических обо-

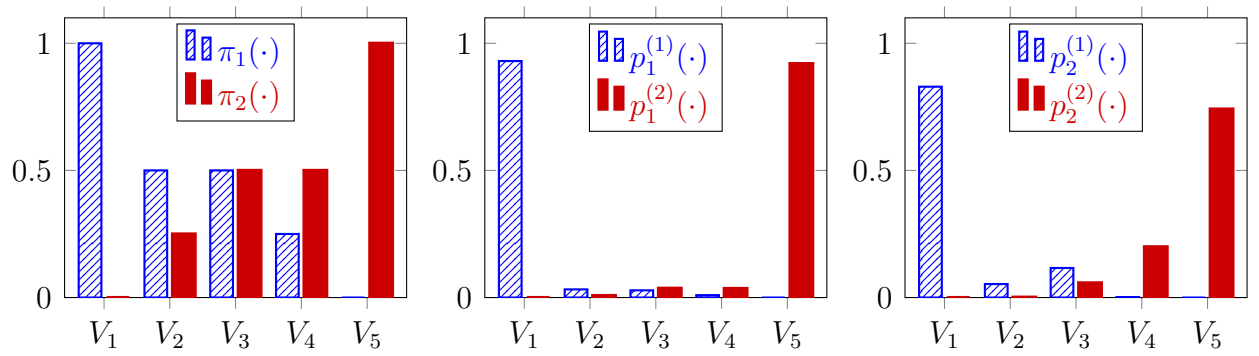


Рисунок 3 — Распределения $\pi_i(\cdot)$ и $p_l^{(i)}(\cdot)$ возможностей Π_i и вероятностей $P_l^{(i)}$, $i, l = 1, 2$, определяющих возможные модели $(\mathbf{V}, \mathcal{A}, \Pi_1)$, $(\mathbf{V}, \mathcal{A}, \Pi_2)$ случайных форм изображений, максимально согласованные в совокупности как с вероятностными моделями $(\mathbf{V}, \mathcal{A}, P_1^{(1)})$, $(\mathbf{V}, \mathcal{A}, P_1^{(2)})$, так и с $(\mathbf{V}, \mathcal{A}, P_2^{(1)})$, $(\mathbf{V}, \mathcal{A}, P_2^{(2)})$. Здесь $\mathbf{V} = \{V_1, \dots, V_5\}$, $\mathcal{A} = 2^{\mathbf{V}}$.

значений, встречающихся во многих формулах документа) определить собственные команды (макросы). Например, вместо того, чтобы везде в тексте набирать громоздкую конструкцию

```
1 F_\text{\rm LDR}
```

можно определить краткую команду `\FLDR`:

```
1 \newcommand{\FLDR}{F_\text{\rm LDR}}
```

и далее везде в тексте использовать её.

Как правило, определение собственных команд производят в *преамбуле* документа, т. е. до строки

```
1 \begin{document}
```

Но это не является обязательным требованием.

2.5 Набор теорем, лемм, следствий, доказательств, определений, примеров и замечаний

Для набора теорем, лемм, следствий, доказательств, определений, примеров и замечаний в пакете NeuroFuzzy определены окружения `theorem`, `lemm`, `conseq`, `proof`, `definition`, `example` и `notice` соответственно. Их использование имеет следующий вид:

```
1 \begin{theorem}
2   Текст теоремы.
```

```

3 \end{theorem}
4 \begin{proof}
5     Доказательство
6 \end{proof}

```

В качестве примера см. теорему 1.

Теорема 1. $F_{\text{LDR}}(\pi) \subset F_*(\pi)$.

Доказательство. Assume that $f \in F_{\text{LDR}}(\pi)$ but $f \notin F_*(\pi)$. Then $\exists g \in F$ such that $\max \{ \pi(s) \mid u(g(s)) \neq u_{\max}(s) \} < \max \{ \pi(s) \mid u(f(s)) \neq u_{\max}(s) \}$. Therefore, there exists integer i s. t. $u(g(s)) = u_{\max}(s)$, $\forall s \in S_1^\pi \cup \dots \cup S_i^\pi$, while $u(f(s_0)) < u_{\max}(s_0) = u(g(s_0))$ for some $s_0 \in S_i^\pi$. Thus, $g >_\pi f$ which contradicts to the initial assumption. \square

2.6 Оформление списка литературы и ссылок на литературу

Согласно ГОСТ ссылки на литературу даются в квадратных скобках, см. ниже, при этом источники нумеруются в том порядке, в котором они встречаются в основном тексте. Т. е. первый встретившийся источник должен иметь номер 1, второй — номер 2 и т. д. Это противоречит здравому смыслу и правилам, принятым в большинстве международных научных журналов, т. к. делает практически неразрешимой задачу поиска нужного источника (например, когда вам известен автор статьи, и вы хотите найти эту статью) в объёмном списке источников. Но таковы требования ГОСТ.

Ниже в настоящем разделе приведён текст, изобилующий ссылками на литературу, которые будут полезны студентам кафедры математического моделирования и информатики.

Современную аксиоматику теории вероятностей впервые предложил отечественный учёный А. Н. Колмогоров в своей работе [1]. А вот ещё несколько «классических» книг по теории вероятностей: [2—5]. Вот несколько хороших книг по математической статистике: [6—8]. О теории нечётких множеств Л. Заде и направлениях теории возможностей, развиваемых иностранными авторами, поведают замечательные работы [9, 10], а о теории возможностей, развиваемой на кафедре математического моделирования и информатики, — работы [11, 12]. Чтобы узнать о морфологическом анализе изображений Ю. П. Пытьева, см. [13—18].

ЗАКЛЮЧЕНИЕ

В заключении подытоживаются результаты исследования. Делается акцент на основных идеях и результатах работы, изложенных в основной части. Указывается, в каких областях науки и техники и при каких условиях эти идеи и результаты могут быть полезны. Делаются предположения о том, как можно было бы продолжить и развить проведённые исследования.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. *Колмогоров А. Н.* Основные понятия теории вероятностей. — ОНТИ НКТП СССР, 1936. — (Математика в монографиях).
2. *Феллер В.* Введение в теорию вероятностей и её приложения. — М.: Мир, 1984.
3. *Невё Ж.* Математические основы теории вероятностей. — М.: «Мир», 1969.
4. *Ширяев А. Н.* Вероятность. — М.: МЦНМО, 2004.
5. *Пытьев Ю. П., Шишмарёв И. А.* Теория вероятностей, математическая статистика и элементы теории возможностей для физиков. — М.: Физический факультет МГУ им. М. В. Ломоносова, 2010.
6. *Леман Э.* Проверка статистических гипотез. — 1979.
7. *Барра Ж.-Р.* Основные понятия математической статистики. — М.: «Мир», 1974.
8. *Боровков А. А.* Математическая статистика. — Новосибирск: Наука, Издательство Института математики, 1997.
9. *Zadeh L. A.* Fuzzy sets as a basis for a theory of possibility // Fuzzy Sets and Systems. — 1978. — Vol. 1, no. 1. — P. 3–28.
10. *Дюбуа Д., Прад А.* Теория возможностей. Приложения к представлению знаний в информатике. — М.: «Радио и связь», 1990.
11. *Пытьев Ю. П.* Возможность. Элементы теории и применения. — М.: «Эдиториал УРСС», 2000.
12. *Пытьев Ю. П.* Возможность как альтернатива вероятности. Математические и эмпирические основы, применение. — М.: ФИЗМАТЛИТ, 2007.
13. *Пытьев Ю. П.* Морфологические понятия в задачах анализа изображений // Докл. АН СССР. — 1975. — Т. 224, № 6. — С. 1283—1286.
14. *Пытьев Ю. П.* Морфологический анализ изображений // Докл. АН СССР. — 1983. — Т. 269, № 5. — С. 1061—1064.

15. *Пытьев Ю. П.* Задачи морфологического анализа изображений // В сб. ст. «Математические методы исследования природных ресурсов Земли из космоса». — 1984. — С. 41—82.
16. *Пытьев Ю. П., Чуличков А. И.* Методы морфологического анализа изображений. — М.: ФИЗМАТЛИТ, 2010.
17. *Визильтер Ю. В.* Теория и методы морфологического анализа изображений : дис. ... д-ра физ.-мат. наук : 05.13.17 / Визильтер Юрий Валентинович. — ФГУП «ГосНИИАС», 2009.
18. Обработка и анализ изображений в задачах машинного зрения. Курс лекций и практических занятий / Ю. В. Визильтер [и др.]. — М.: Физматкнига, 2010.

ПРИЛОЖЕНИЕ А

Зачем нужны приложения

В приложения выносятся материал, который не имеет принципиального значения для понимания основного текста. Приложения можно разбивать на разделы с помощью команд `\section`, `\subsection` и т. д., как и обычные главы. Ссылки на приложения делаются точно так же, как и на любые другие разделы.

Приложения создают только в том случае, если в этом есть необходимость. Очень часто приложений в курсовых, дипломных и др. работах нет, в этом случае работа оканчивается разделом «Список использованных источников».

ПРИЛОЖЕНИЕ Б

Справочник часто используемых специальных символов и команд ЛАТЭХ

Таблица 2 — Часто используемые специальные символы и команды.

Пример	Исходный текст	Описание
--------	----------------	----------

Пробелы и пунктуация

таблица 1	таблица~1	неразрывный пробел переменной длины
А. В. Зубюк	А.\,В.\;Зубюк	короткий и длинный неразрывные пробелы
физ.-мат. науки	физ.-мат. науки	дефис
стр. 1–10	стр.~1--10	минус или диапазон
тире — длинная черта	тире~--- длинная черта	тире
метро «Университет»	метро <<Университет>>	кавычки-«ёлочки»
next stop is “University”	next stop is ``University''	кавычки-«лапки»

Метки и ссылки

	<code>\label{sec:make}</code>	метка
раздел 1.6	раздел~\ref{sec:make}	ссылка по номеру
стр. 8	стр.~\pageref{sec:make}	ссылка на страницу
раздел «Сборка с использованием утилиты make»	раздел <<\nameref{sec:make}>>	ссылка на заголовок
(1)	<code>\eqref{eq:circle}</code>	ссылка на формулу

Математика

a^i, a_j, A_j^i	<code>a^i, a_j, A^i_j</code>	индексы
α, β, Ω	<code>\alpha, \beta, \Omega</code>	греческие буквы
$\int_0^\infty, \iint, \oint$	<code>\int\limits_{0}^\infty, \iint, \oint</code>	интегралы

Таблица 2 — продолжение.

Пример	Исходный текст	Описание
$\sum_{i=1}^n$	<code>\sum_{i=1}^n</code>	сумма
$\prod_{i=1}^n$	<code>\prod_{i=1}^n</code>	произведение
$\frac{a}{b}, \frac{a}{b}$	<code>\frac{a}{b},</code> <code>\dfrac{a}{b}</code>	дроби
$\sin, \exp, \partial, \nabla, \sim, \approx, \in,$ $\subset, \not\subset, \min_a, \max_a, \inf_a, \sup_a,$ $\lim_{a \rightarrow b}$	<code>\sin, \exp, \partial,</code> <code>\nabla, \sim, \approx,</code> <code>\in, \subset, \notin,</code> <code>\min_a, \max_a, \inf_a,</code> <code>\sup_a, \lim_{a \to b}</code>	математические функции, операторы и отношения
$\vec{a}, \tilde{a}, \hat{a}, \widetilde{abc}$	<code>\vec a, \tilde a,</code> <code>\hat a, \widetilde{abc}</code>	векторы и другое декорирование
$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$	<code>\begin{pmatrix} 1 & 2 & \\ 3 & 4 & 5 & 6 \end{pmatrix}</code> <code>\end{pmatrix}</code>	матрица
$\xrightarrow[n \rightarrow \infty]{\text{п. н.}}, \rightarrow, \Rightarrow, \Leftarrow$	<code>\xrightarrow[n \to \infty]</code> <code>{\text{п. н.}}, \to,</code> <code>\Rightarrow, \Leftarrow</code>	стрелки, текст в формуле
$ a , \ a\ , \left(1 + \frac{a}{b}\right)$	<code> a , \ a\ , \left(1 +</code> <code>\frac{a}{b}\right)</code>	модуль, норма, большие скобки
$x = 0$	<code>\[x=0 \]</code>	обособленная формула
$x = 0 \quad (1)$	<code>\begin{equation} x=0</code> <code>\end{equation}</code>	нумерованная формула
$\delta_{ij} = \begin{cases} 0 \\ 1 \end{cases}$	<code>\delta_{ij} =</code> <code>\begin{cases} 0 \\ 1 \end{cases}</code> <code>\end{cases}</code>	варианты, система уравнений
$S = \pi r^2,$ $l = 2\pi r.$	<code>\begin{gather*} S = \pi</code> <code>r^2, \\ l = 2\pi r.</code> <code>\end{gather*}</code>	объединение формул

Таблица 2 — продолжение.

Пример	Исходный текст	Описание
$S = \pi r^2 =$ $= \pi d^2/4$	<pre>\begin{multline*} S =</pre> $\pi r^2 = \pi d^2/4$ <pre>\end{multline*}</pre>	<p>многострочная формула</p>