

1. Построение алфавитного указателя

Подготовка алфавитного указателя — весьма длительная процедура. Она включает разметку терминов по всему исходному файлу и двукратный прогон L^AT_EX’а, причем относящиеся к указателю данные, полученные при первом прогоне, должны быть обработаны программой `MakIndex`.

Процедура подготовки алфавитного указателя состоит из следующих шагов.

- В команду `\documentstyle` следует ввести корректирующий стиль `makeidx`.
- В преамбулу документа следует ввести команду `\makeindex`.
- Каждый термин, который должен быть помещен в алфавитный указатель, определяется командой `\index` с соответствующим аргументом.
- В месте выдачи алфавитного указателя следует поместить команду `\printindex`.
- Подготовленный файл обработать с помощью L^AT_EX’а. При этом будет создан файл с именем исходного файла и расширением `idx`, содержащий все помеченные командой `\index` термины.
- Обработать `idx`-файл программой `MakIndex`, вызов которой есть

```
MAKEINDEX <имя файла>
```

Программа создаст файл `<имя файла>.ind` и протокольный файл `<имя файла>.ilg`.

- Еще раз обработав исходный файл L^AT_EX’ом, вы получите документ, содержащий алфавитный указатель. Алфавитный указатель всегда начинается с новой страницы.

Несколько команд `\index` с одним и тем же аргументом на одной странице не влияют на построение указателя, поэтому термин, который должен попасть в индекс, можно каждый раз вводить в сопровождении команды `\index` (например, не просто `tabular`, а `tabular\index{tabular}`).

Аргумент команды `\index` имеет несколько форматов.

- ▷ `\index{<вход>}`
Вход в алфавитном указателе будет ссылаться на номера страниц.
- ▷ `\index{<вход>|{}` % на первой странице области
- ▷ `\index{<вход>|)}` % на последней странице области
Вход в алфавитном указателе будет ссылаться на область страниц.

- ▷ `\index{<вход>!<субвход>}`
Вход в алфавитном указателе будет иметь субвход первого уровня.
- ▷ `\index{<вход>!<субвход>!<субсубвход>}`
Вход в алфавитном указателе будет иметь субвход второго уровня.
- ▷ `\index{<вход1>!<субвход1>|see{<вход, субвход>}}`
- ▷ `\index{<вход1>|see{<вход>}}`
Вход в алфавитном указателе будет ссылаться на другой вход.
- ▷ `\index{<ключ>@<вход>}`
Последовательность символов `<ключ>` является ключом для сортировки в алфавитном порядке и определяет место `<входа>`.
- ▷ `\index{<вход>|<имя команды>}`
Чтобы ввести особый формат выдачи номера страницы (например, номер страницы, где термин встречается впервые, желательно выделить курсивом), в аргументе команды `\index` следует после разделителя `|` указать имя команды форматирования без символа `\`.

Чтобы поместить в аргумент команды `\index` символы `!`, `@`, `|` как обычные символы, перед каждым из них следует выдать символ двойных кавычек.

Команды `\index`, аргументы которых состоят из одних и тех же символов, но по-разному разделенных пробелами, создают в алфавитном указателе различные входы.

2. Использование баз данных при составлении библиографии

Большой список литературы или список, основная часть которого интенсивно используется, целесообразно хранить в базе данных.

Очевидное преимущество работы с базами данных состоит в том, что библиографическую информацию вы набираете только один раз, при занесении ее в базу данных. Каждый пункт библиографии оформляется как отдельная запись с уникальной меткой, и впоследствии вставка этого пункта в список литературы любого документа со ссылкой на него в тексте или без ссылки выполняется простыми по формату командами, указывающими эту метку.

Составление списка литературы осуществляется с помощью двух описательных команд, одна из которых определяет стиль библиографического списка, а вторая — место расположения списка (это место выдачи данной команды) и используемые базы данных.

2.1. Команды работы с библиографией, хранящейся в базе данных

Извлечение информации в конкретную публикацию выполняется посредством уже известной вам команды `\cite` и команды `\nocite`. В качестве обязательных параметров этих команд используются метки записей библиографических баз данных.

▷ `\cite[<текст>]{<список меток>}`

Команда ссылки на один или несколько пунктов списка литературы. Обязательный параметр должен содержать одну или несколько меток, разделенных запятыми или тире. Необязательный параметр `<текст>` будет представлен как комментарий к пункту списка литературы.

▷ `\nocite[<текст>]{<список меток>}`

Команда указывает библиографическую информацию, на которую нет ссылок в тексте, но которую следует ввести в список литературы. Обязательный параметр должен содержать одну или несколько меток, разделенных запятыми. Необязательный параметр `<текст>` будет представлен как комментарий к пункту списка литературы.

При работе с библиографическими базами данных вместо использования командных скобок `\thebibliography` используются две команды. Первая из них задает *стиль* библиографии, вторая указывает базы данных, из которых будут выбираться требуемые пункты (те, чьи метки указаны в документе в командах `\cite` и `\nocite`).

▷ `\bibliographystyle{<стиль>}`

задает стиль библиографии, который определяет вид списка литературы — как стиль документа определяет формат документа. Команда `\bibliographystyle` должна появляться *после* команды `\begin{document}`.

Файлы описания библиографических стилей имеют расширение `.bst`. Доступны следующие стандартные библиографические стили:

`plain` элементы отсортированы по алфавиту и помечены числами, имена и фамилии авторов указаны полностью;

<code>unsrt</code>	порядок расположения элементов соответствует расположению ссылок в тексте, имена и фамилии авторов указаны полностью;
<code>alpha</code>	элементы отсортированы по алфавиту и помечены символьными метками, как правило составленными с использованием имени автора и года публикации;
<code>abbrv</code>	элементы отсортированы по алфавиту и помечены числами, фамилии авторов, название месяца и журнала даны в сокращенном виде.

▷ `\bibliography{<файл1>,<файл2>...}`

Обязательный параметр содержит имена (или имя) файлов используемых библиографических баз данных. Имена файлов следует указывать *без расширения*.

Л^AT_EX позволяет использовать одновременно несколько баз данных.

2.2. Процедура получения списка литературы из библиографических баз данных

Процедура получения библиографии из файлов баз данных состоит из нескольких шагов.

- При обработке документа Л^AT_EX'ом создается рабочий файл с расширением `aux`, который будет содержать все ключи для ссылок и все аргументы команд `\bibliographystyle`, `\cite` и `\nocite`, `\bibliography`. Этот файл следует обрабатывать программой `VibTeX`, вызываемой командой

`VIBTEX <имя файла без расширения>`

Программа читает информацию из файла `aux` и создает одноименный файл с расширением `bbl`, содержащий команды Л^AT_EX'а для получения списка литературы. Одновременно создается протокольный файл, имеющий расширение `blg`.

- При повторной обработке исходного файла (после работы программы `VibTeX`), Л^AT_EX читает `bbl`-файл (по команде `\bibliography`) и вставляет в текст список литературы.
- Наконец при третьем прогоне Л^AT_EX выполнит расстановку ссылок.

2.3. Формат файла библиографических данных

База данных — это файл, разделенный на записи. Каждая запись хранит библиографическую информацию о какой-либо одной работе.

Файл базы данных для Л^AT_EX'а готовится любым редактором. Имена файлов библиографических баз данных должны иметь расширение `bib`.

Запись начинается символом `@`. За ним следует указание *типа* работы, которую представляет запись — `article` (статья), `book` (книга) и прочее.

За типом записи в фигурных скобках следует собственно информация о работе. Она состоит из *идентификатора* и *полей*.

Идентификатор есть уникальная метка записи. Именно он должен быть использован в качестве обязательного аргумента команды `\cite` или `\nocite` при ссылке на данную работу.

—
—
Каждое поле состоит из имени, знака равенства и текстовой строки, заключенной в фигурные скобки или двойные кавычки. Если текст полностью состоит из цифр, скобки или кавычки могут быть опущены.

Поля классифицируются следующим образом:

- обязательное* отсутствие поля вызовет сообщение об ошибке;
- необязательное* информация в этом поле будет использована, если она присутствует в записи, но ее отсутствие не вызовет сообщения об ошибке и каких-либо проблем при печати. Некоторые библиографические стили могут просто оставлять без внимания определенные поля;
- игнорируемое* если BibTeX не распознает поле как обязательное или необязательное, то это поле игнорируется. Поэтому пользователь может включать в запись любые поля, для хранения нужной ему информации (например, ANNOTATION — поле, в котором можно хранить аннотацию работы).

Поля разделяются запятыми. Порядок их ввода не существен.

BibTeX не делает различия между прописными и строчными буквами в идентификаторах записей или названиях полей.

2.3.1. Типы записей

Здесь перечислены основные типы записей с указанием их *обязательных* и *необязательных* полей, которые используются в стандартных библиографических стилях.

- article — журнальная статья.

Обязательные поля

author (автор)
title (заголовок)
journal (журнал)
year (год)

Необязательные поля

volume (том)
number (номер)
pages (страницы)
month (месяц)
note (примечание)

- book — книга.

Обязательные поля

author (автор) / editor (редактор)
title (заголовок)
publisher (издательство)
year (год)

Необязательные поля

volume (том)
series (серия)
address (адрес)
edition (издание)
month (месяц)
note (примечание)

- booklet — буклет (печатная работа в переплете без ссылки на издательство).

Обязательные поля

`title` (заголовок)

Необязательные поля

`author` (автор)

`howpublished` (способ издания)

`address` (адрес)

`month` (месяц)

`year` (год)

`note` (примечание)

- `conference` — конференция. См. ниже `inproceedings`.

- `inbook` — выдержка из книги.

Обязательные поля

`author` (автор) / `editor` (редактор)

`title` (заголовок)

`publisher` (издательство)

`chapter` (глава) и/или `pages` (страницы)

`year` (год)

Необязательные поля

`volume` (том)

`series` (серия)

`address` (адрес)

`edition` (издание)

`month` (месяц)

`note` (примечание)

- `incollection` — выдержка из книги с заголовком.

Обязательные поля

`author` (автор)

`title` (заголовок)

`booktitle` (заголовок книги)

`publisher` (издательство)

`year` (год)

Необязательные поля

`editor` (редактор)

`chapter` (глава)

`pages` (страницы)

`address` (адрес)

`month` (месяц)

`note` (примечание)

- `inproceedings` — статья в трудах конференции.

Обязательные поля

`author` (автор)

`title` (заголовок)

`booktitle` (общий заголовок)

`year` (год)

Необязательные поля

`editor` (редактор)

`organization` (организация-спонсор)

`pages` (страницы)

`publisher` (издательство)

`address` (адрес)

`month` (месяц)

`note` (примечание)

- `manual` — техническая документация.

Обязательные поля

`title` (заголовок)

- `masterthesis` — диссертация магистра.

Обязательные поля

`author` (автор)

`title` (заголовок)

`school` (учебное заведение)

`year` (год)

- `misc` — разное.

Обязательные поля

- `phdthesis` — докторская диссертация.

Обязательные поля

`author` (автор)

`title` (заголовок)

`school` (учебное заведение)

`year` (год)

- `proceedings` — труды конференции.

Обязательные поля

`title` (заголовок)

`year` (год)

- `techreport` — отчет, который издается учебным или каким-либо другим учреждением и обычно нумерован по сериям.

Необязательные поля

`author` (автор)

`organization` (организация)

`edition` (издание)

`address` (адрес)

`year` (год)

`month` (месяц)

`note` (примечание)

Необязательные поля

`address` (адрес)

`month` (месяц)

`note` (примечание)

Необязательные поля

`author` (автор)

`title` (заголовок)

`howpublished` (способ издания)

`year` (год)

`month` (месяц)

`note` (примечание)

Необязательные поля

`address` (адрес)

`month` (месяц)

`note` (примечание)

Необязательные поля

`editor` (редактор)

`publisher` (издательство)

`organization` (организация-спонсор)

`address` (адрес)

`month` (месяц)

`note` (примечание).

Обязательные поля

`author` (автор)
`title` (заголовок)
`institution` (учреждение)
`year` (год)

Необязательные поля

`type` (тип отчета)
`number` (номер)
`month` (месяц)
`note` (примечание)

- `unpublished` — не опубликованный официально документ, имеющий тем не менее имя автора и название.

Обязательные поля

`author` (автор)
`title` (заголовок)
`note` (примечание)

Необязательные поля

`month` (месяц)
`year` (год)

В добавление к вышперечисленным полям, каждый тип записи может содержать еще и необязательное поле `key`, которое используется некоторыми стилями для сортировки.

2.3.2. Преамбула

База данных может начинаться с особого типа записи — преамбулы. В этой записи вы можете определить средствами ЛАТ_EX'a новые макрокоманды, которые облегчат построение конкретного списка литературы.

Определения новых команд помещаются в общие групповые скобки. Каждое определение помещается в двойные кавычки. Друг от друга определения отделяются символом `#`.

```
@preamble{"\newcommand{\printfirst}[2]{#1} "  
# "\def\n#1{\hbox{#1}}"  
# "\newcommand{\singleletter}[1]{#1} "  
# "\newcommand{\switchargs}[2]{#2#1} " }
```

2.3.3. Символьные последовательности

Еще один допустимый тип записи предназначен для облегчения ввода повторяющихся последовательностей символов, например, информации об авторе многих включенных в базу данных работ.

Аргумент записи помещается в круглые скобки и содержит название последовательности и ее значение, соединенные знаком равенства. Значение последовательности должно быть заключено в двойные кавычки.

```
@STRING( Klímenko = "Станислав Владимирович Клименко")
```


2.3.4. Ввод информации об авторах и редакторах

При вводе информации в поля `author` и `editor` важно соблюдать следующие правила.

- Для записи полных данных — фамилии, имени и, возможно, отчества — существует два формата:

```
"Donald Ervin Knuth"           "Knuth, Donald Ervin"  
"Станислав Владимирович Клименко"  "Клименко, Станислав Владимирович"
```

В первом формате, в котором отсутствует запятая, фамилией считается последнее слово, а предыдущие будут введены в список литературы перед фамилией в виде начальных букв и точек за ними. При втором формате фамилией считается слово, отделенное запятой, а остальные слова будут представлены лишь инициалами.

- Если в базу данных вводятся фамилии и инициалы авторов, то между инициалами должен быть пробел.

```
"С. В. Клименко"           "Клименко, С. В."
```

- Если авторов несколько, между их именами следует вставлять символы `and`.
- Если авторов очень много, список их можно прервать словами `and others`, которые в списке литературы дадут стандартное обозначение `et al.`

2.4. Несколько замечаний

- Материал нашей книги по использованию библиографических баз данных дает лишь общее знакомство с простейшей техникой работы с ними. Для изучения всех возможностей программы `BibTeX` следует ознакомиться с технической документацией по этой программе.
- В настоящее время нам неизвестен законченный программный продукт, представляющий русифицированные библиографические стилевые файлы с учетом особенностей составления списков литературы в публикациях российских журналов. Поэтому в представленных здесь примерах участвуют лишь простейшие пункты литературы на русском языке.

В качестве примера приводим часть базы данных, являющейся основой списка литературы нашего руководства. Стилиевой файл компоновки данных из базы данных в список литературы был модифицирован для именно нашего случая.

```
@preamble{ "\newcommand{\noopsort}[1]{} "  
           # "\newcommand{\printfirst}[2]{#1} "  
           # "\def\@{,} "  
           # "\def\n#1{\hbox{#1}}"  
           # "\newcommand{\singleletter}[1]{#1} "  
           # "\newcommand{\switchargs}[2]{#2#1} " }  
  
@STRING( IHEP      = "ИФВЭ")  
@STRING( Protvino = "Протвино")
```

```

@STRING( Gritsaenko = "Ирина Анатольевна Грицаенко")
@STRING( Klimenko = "Станислав Владимирович Клименко")
@STRING( Samarin = "Александр Валимович Самарин")
@STRING( Malyshev = "Василий Константинович Малышев")
@STRING( Glonty = "Нана Левановна Глonti")

@PREPRINT{ihep-tex,
  author = Gritsaenko # " and " # Klimenko # " and " # Malyshev
          # " and " # Samarin,
  title = "{\TeX} в {\ИФВЭ}. 1.~Общая характеристика",
  institution = IHEP,
  address      = Protvino,
  year        = 1991,
  number      = 54,
  pages       = 38,}

@ARTICLE{rustex,
  author = "B.K. Malyshev and A.V. Samarin and D. Vulys ",
  title  = "Russian {\TeX}",
  year   = 1991,
  journal = "TUGBoat 12(2)", pages = 3,}

@INPROCEEDINGS{fr-rustex,
  author = "B.K. Malyshev and A.V. Samarin and D. Vulys",
  title  = "Russian {\TeX}",
  booktitle = "Proceedings of 6th EUROTEX conference",
  address  = "Paris",
  year     = 1991,
  pages    = "3-10",
  numpages = 8,}

@BOOK{texbook,
  author = "Donald Ervin Knuth",
  title  = "The \TeX book",
  publisher = "Addison-Wesley", address = "Reading, Massachusetts",
  year = "1984",}

@BOOK{latexbook,
  author = "Lesly Lamport",
  title  = "\LaTeX: a Document Preparations System",
  publisher = "Addison-Wesley",
  address  = "Reading, Massachusetts",
  year = "1986",}

```

3. Вставка иллюстраций

Внешняя по отношению к подготавливаемому средствами $\text{T}_\text{E}\text{X}$ 'а или $\text{L}\text{A}\text{T}_\text{E}\text{X}$ 'а графика — это иллюстрации, запакованные в файл в формате языка какого-либо внешнего устройства.

Драйверы семейства DVIDRV коллекции $\text{e}\text{m}\text{T}_\text{E}\text{X}$, работающие под контролем операционной системы MS-DOS на IBM-PC, оперируют лишь с графикой в формате PCX.

3.1. Вставка картинок примитивом $\backslash\text{special}$

Фундаментальное средство предоставления системе внешней графики есть примитив $\text{T}_\text{E}\text{X}$ 'а:

```
 $\backslash\text{special}$  {<метка> <имя файла>}
```

Обязательный параметр <метка> как правило состоит из двух частей: первая часть указывает драйвер, т.к. драйверы для различных внешних устройств отнюдь не совместимы, вторая часть метки указывает этому драйверу, что в данном месте документа должен быть вставлен внешний файл. Следующий обязательный параметр указывает имя этого файла.

Драйверы семейства DVIDRV коллекции $\text{e}\text{m}\text{T}_\text{E}\text{X}$ воспринимают следующий формат этого примитива:

```
 $\backslash\text{special}$  {em:graph <имя файла>}
```

3.2. Вставка растровых картинок в формате PCX средствами $\text{L}\text{A}\text{T}_\text{E}\text{X}$ 'а

$\text{L}\text{A}\text{T}_\text{E}\text{X}$ в большой степени унифицирует процедуру вставки картинок в разных форматах для различных драйверов. Он использует информацию о размерах иллюстрации, имеющуюся во внешних файлах, и не нуждается в отведении места для картинки дополнительной командой. $\text{L}\text{A}\text{T}_\text{E}\text{X}$ позволяет легко вводить рамку вокруг рисунка, а также смещать ее отдельные стороны или двигать рисунок в ее пределах для получения желаемого визуального эффекта.

При вставке в текст иллюстрации в формате PCX *не могут* быть масштабированы.

Для вставки иллюстраций в формате PCX в команде $\backslash\text{documentstyle}$ следует задать корректирующий стиль BitMap .

3.2.1. Задание метрической информации

Метрическая информация, т.е. информация о размерах картинке, может находиться в отдельном файле с тем же именем, что и содержащий картинку файл, и расширением trm . Этот файл создается программой pcxsize и имеет следующий вид:

```

\def\xbegpixel{<llx>}% левый нижний X
\def\ybegpixel{<lly>}% левый нижний Y
\def\xendpixel{<urx>}% правый верхний X
\def\yendpixel{<ury>}% правый верхний Y

```

где $\langle llx \rangle$, $\langle lly \rangle$, $\langle urx \rangle$, $\langle ury \rangle$ есть соответственно x и y координаты нижнего левого и верхнего правого углов ограничивающего картинку прямоугольника, задаваемые в пикселях.

Если размеры картинки известны, этот файл можно создать редактором.

При вставке иллюстрации в формате РСХ метрический файл не является обязательным. Размеры картинки можно указать как параметр непосредственно в команде вставки иллюстрации. Однако этот файл удобен хотя бы как хранилище информации, и лучше ввести его построение в качестве стандарта.

Размеры ограничивающего картинку прямоугольника специально задаются в пикселях, чтобы можно было оценить размеры иллюстрации при выводе на печатающие устройства с различным разрешением.

3.2.2. Команды вставки иллюстраций

▷ `\BitMap[llx lly urx ury]{<файл>}`

Команда вставки иллюстрации в формате РСХ. Обязательный параметр `<файл>` указывает имя и расширение вставляемого файла. Необязательный параметр задает размеры картинки:

координаты (llx, lly) левого нижнего угла ограничивающего ее прямоугольника,
координаты (urx, ury) правого верхнего угла этого прямоугольника.

Если этот параметр отсутствует, информация о размерах картинки будет взята из метрического файла. По умолчанию картинка будет центрирована относительно строки, в которой выдана и вокруг картинки будет проведена рамка,

Для выравнивания по текстовой строке верхнего или нижнего края картинки, а также центрирования ее относительно этой строки существуют команды:

▷ `\tBitMap{<файл>}`

Вставляет картинку, помещая в месте выдачи команды левый верхний угол картинки.

▷ `\bBitMap{<файл>}`

Вставляет картинку, помещая в месте выдачи команды левый нижний угол картинки.

▷ `\cBitMap{<файл>}`

Вставляет картинку, центрируя картинку относительно текстовой строки (принято по умолчанию).

3.2.3. Команды работы с рамкой

Параметры команд работы с рамкой должны иметь размерность, распознаваемую ЛАТ_EX'ом. Положительное значение величины приводит к сдвигу соответствующей стороны к центру прямоугольника. Эти команды действуют только на *одну* картинку:

- ▷ `\TrimBoundingBox{<величина>}`
Изменяет размер отведенного для картинки бокса путем сдвига всех его сторон на одну и ту же величину.
- ▷ `\TrimTop{<величина>}` % верх
Команда сдвига верхней стороны отведенного для картинки бокса.
- ▷ `\TrimLeft{<величина>}` % левая сторона
Команда сдвига левой стороны отведенного для картинки бокса.
- ▷ `\TrimBottom{<величина>}` % низ
Команда сдвига нижней стороны отведенного для картинки бокса.
- ▷ `\TrimRight{<величина>}` % правая сторона
Команды сдвига правой стороны отведенного для картинки бокса.

Команды, убирающие и восстанавливающие рамку. Действуют на все дальнейшие иллюстрации вплоть до выдачи новой команды этого типа:

- ▷ `\HideDisplacementBoxes` % скрыть
Устанавливает вывод картинок без рамок.
- ▷ `\ShowDisplacementBoxes` % показать
Устанавливает вывод картинок с рамками. Принята по умолчанию.

Команды сдвига картинки без изменения положения ее бокса. Их параметры должны иметь размерность, распознаваемую ЛАТ_EX'ом. Эти команды действуют только на *одну* картинку.

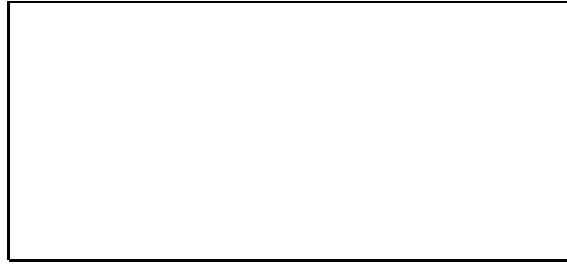
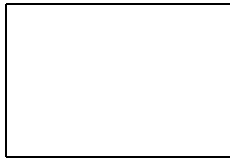
- ▷ `\hSlide{<величина>}` % горизонтальный сдвиг
Положительное значение параметра сдвигает картинку вправо, а отрицательное значение — влево.
- ▷ `\vSlide{<величина>}` % вертикальный сдвиг
Положительное значение параметра сдвигает картинку вверх, а отрицательное значение — вниз.

Картинка в формате РСХ находится в файле `ig2.pcx`. Запуск программы `pcxsize` командой

```
PCXSIZE IG2.PCX
```

позволил определить размеры картинки и поместить их в файл `trm`.

Сначала картинка печатается «как есть», без перемещений рамки или ее самой. Далее мы помещаем две картинки в одной рамке, сдвинутых относительно друг друга по вертикали. При этом использовались команды смещения сторон рамки, команды ввода положительного и отрицательного горизонтальных промежутков и команды смещения самой картинки.



```
\BitMap{ig2.pcx} \hspace*{1cm}  
\TrimTop{-7mm} \TrimBottom{-7mm} \TrimLeft{-40mm} \TrimRight{-5mm}  
\vslide{4mm} \BitMap{ig2} \hspace*{-60mm} \HideDisplacementBoxes  
\vSlide{-4mm}\hSlide{-13mm} \BitMap{ig2}
```

Наконец, следующие «административные» команды бывают полезны:

- ▷ `\BitMapResolution{<число>}{<число>}`
Задаёт разрешение печатающего устройства. По умолчанию как по горизонтали, так и по вертикали принято разрешение 300 точек на дюйм.
- ▷ `\BitMapDirectory{<каталог>}`
Указывает каталог, в котором хранится файл с картинкой (если он не находится в текущем каталоге).

4. Шрифты

Основные группы команд работы со шрифтами:

- команды, декларирующие типы и размеры нематематических шрифтов;
- команды ввода множества математических символов

— рассмотрены соответственно в разделах, описывающих ввод простого текста и работу с математическими выражениями.

Все эти команды формализованы так, что пользователь не нуждается в информации, где и как хранятся шрифты и как осуществляется доступ к ним.

Параметры команд задания размеров шрифта зависят от выбора нормального шрифта документа, и вычисляются в соответствующих файлах описания стиля, а команды декларации типа шрифта зависят от конфигурации схемы выбора шрифтов, которая была принята при текущей компиляции системы (об этом можно узнать подробнее в разделе технической документации, посвященном новой схеме выбора шрифтов).

Однако данного набора команд может быть недостаточно для эффективной работы по следующим причинам:

- коллекция шрифтов велика и постоянно расширяется, и не все шрифты охватываются вышеуказанными командами;
- пользователя может не устраивать выбор шрифтов, делаемый по умолчанию в файлах описания стилей.

TeX и LaTeX предоставляют средства для выбора и загрузки любого шрифта и извлечения любого символа этого шрифта, позволяя пользователю “напрямую” работать с со всей коллекцией шрифтов.

Для такого “прямого” общения пользователю надо знать, как устроена библиотека шрифтов.

4.1. Классификация шрифтов

Шрифты объединяются в семейства — аналоги гарнитур классической полиграфии.

Каждое семейство имеет краткое название, используемое в командах описания и загрузки шрифтов. Например, `cmr` — название семейства шрифтов Computer Modern Roman, разработанного Д.Кнудом и являющегося важной компонентой TeX'a.

Семейства весьма разнообразны.

В семейства, которые наиболее часто используются при подготовке публикаций, входят шрифты различных видов — нормальные, жирные, курсивные и некоторые иные, и каждый вид представлен набором шрифтов различных размеров.

Семейство может состоять и только из одного шрифта — таковы, например, семейства экзотических шрифтов.

- **Текстовые шрифты Computer Modern.**

Три семейства содержат более пятидесяти оригинальных шрифтов и 47 шрифтов русского (подмножество кириллического) алфавита, созданных на графической основе латинских шрифтов.

Названия кириллических шрифтов построены по аналогии с латинскими, но Computer Modern заменено на Computer Modern Cyrillic.

Подмножество кириллических шрифтов системы содержит только буквы русского алфавита и символы № « » (знак номера и кавычки, используемые в публикациях на русском языке).

- `cmr` Computer Modern Roman — семейство шрифтов, базовое для большинства подготавливаемых на \TeX 'е документов. Исторически сложилось так, что в названия многих шрифтов этого семейства входят лишь две первые буквы его аббревиатуры;
- `cmcyr` Computer Modern Cyrillic — подмножество кириллических шрифтов в семействе Computer Modern Roman.
- `cmtt` Computer Modern Typewriter — семейство равноширинных шрифтов;
- `cmctt` Computer Modern Cyrillic Typewriter — подмножество кириллических шрифтов в семействе равноширинных шрифтов.
- `cmss` Computer Modern Sans Serif — семейство рубленых (без засечек) шрифтов.
- `cmcss` Computer Modern Cyrillic Sans Serif — подмножество кириллических шрифтов в семействе рубленых шрифтов.

- **Математические шрифты Computer Modern.**

Математические шрифты разделены на два семейства и включают практически полный набор символов, необходимых для построения математических формул.

- `cmi` Math italic — шрифты с буквами греческого и латинского алфавитов курсивного начертания, цифрами и некоторыми специальными символами;
- `cmsy` Math symbols — шрифты с математическими символами.

- **Экзотические шрифты Computer Modern.**

- `cmf` Funny и Funny Italic;
- `cmtext` \TeX Extended ASCII Type;
- `cmdunh` Dunhill Roman Type.

- **\LaTeX - \TeX шрифты.**

Шрифты, разработанные и поддерживаемые AMS, спонсором \TeX 'а и METAFONT'а на ранних этапах их развития. Эти

—
—
шрифты содержат множество оригинальных математических символов, дополняющих шрифты \TeX 'а.

Шрифты \mathcal{AMS} - \TeX делятся на следующие семейства:

- Euler — три родственных семейства шрифтов. Эти шрифты разработал Германн Запф, и они принадлежат Станфордскому университету как часть \TeX -проекта.
 - euf Euler Fraktur (German) — содержит буквы и иные символы готического начертания;
 - eur Euler Roman (cursive) — содержит буквы греческого и латинского алфавитов прямого курсивного начертания;
 - eus Euler Script — содержит буквы латинского алфавита каллиграфического начертания.
- Шрифты с математическими символами:
 - msa AMS symbol font A;
 - msb AMS symbol font B.
- \mathcal{WNSyr} Cyrillic fonts — семейство шрифтов кириллического алфавита. Разработано в Университете штата Вашингтон (г.Сиэтл). Содержат практически полный алфавит для большинства славянских языков.

Имя шрифта, принадлежащего к определенному семейству, идентифицирует его относительно других шрифтов, имеющих в библиотеке.

По возможности имя шрифта строится определенному формату: первые две–четыре буквы представляют собой аббревиатуру семейства, затем следуют буквы, указывающие по возможности на начертание и форму конкретного шрифта. Цифры в конце — это размер в типографских пунктах. Например, имя шрифта «cmtt12» является аббревиатурой для «Computer Modern Typewriter Type размером в 12 пунктов».

4.2. Простейшие команды выбора шрифта и отдельного символа

Простейший формат команды \TeX 'а загрузки шрифта имеет вид

```
\font\<имя> = <требуемый шрифт>
```

где $\backslash\langle\text{имя}\rangle$ — новая команда, ассоциированная с данным шрифтом, а параметр $\langle\text{требуемый шрифт}\rangle$ — библиотечное название шрифта.

Загрузка шрифта выполняется и по команде \LaTeX 'а

```
\newfont{\<имя>}{<имя шрифта>}
```

В ней определяется имя команды для выбора шрифта, библиотечное название которого указано во втором параметре. (Эта вновь определенная команда не должна использоваться в математической моде.)

Печать отдельного символа текущего шрифта выполняется по команде \TeX 'а

$\backslash\text{char}\langle\text{номер}\rangle$

или команде \LaTeX 'а

$\backslash\text{symbol}\{\langle\text{номер}\rangle\}$

где $\langle\text{номер}\rangle$ есть числовой код, который определяется по таблицам.

abcde	12345	אבגדע	abcde	12345	אבגדע	ABCDE
abcde	12345	ABCDE	ΔΘΛΞΠ	@		

```
\font\eufrb=eufb8      \font\eufrb=eufm10  \font\eufrb=eusb10
                        \newfont{\cmdunh}{cmdunh10}
\par{\eufrb abcde \quad 12345 \quad ABCDE}\quad
      {\eufrb abcde \quad 12345 \quad ABCDE}\quad
      {\eufrb \char65\char66\char67\char68\char69}
\par\medskip {\cmdunh abcde \quad 12345 \quad ABCDE}\quad
              \symbol{1}\symbol{2}\symbol{3}\symbol{4}\symbol{5}\quad\symbol{64}}
```

Чтобы получить таблицу, представляющую символы шрифта и их коды, следует запустить \TeX для обработки системного файла `testfont.tex` командой

```
TEX testfont
```

Будет затребовано название шрифта, а затем тип операции. На последний запрос следует ввести команду

```
\table\end
```

В результате появится содержащий таблицу файл `testfont.dvi`, с которым можно работать как с любым `dvi` файлом.

4.3. Как использовать в математической моде шрифты коллекции $\mathcal{A}\mathcal{M}\mathcal{S}$ - TEX

Шрифты коллекции $\mathcal{A}\mathcal{M}\mathcal{S}$ - TEX не загружаются автоматически при запуске TEX 'а. Их необходимо определить в явном виде, например, командами

```
\font\msam = msam10
\font\msbm = msbm10
```

где $\backslash\text{msam}$ и $\backslash\text{msbm}$ есть логические имена шрифтов, которые пользователь может выбрать по своему усмотрению.

Для TEX 'а шрифты эти не определены как математические, поэтому нельзя использовать их символы в математических выражениях, только указывая порядковый номер символа в шрифте. Например, команды

```
{\msam{\char1}} {\msam{\char39}} {\msbm{\char26}} {\msbm{\char126}}
```

приведут к выдаче желаемых символов $\boxplus \gtrsim \lesssim \hbar$, тогда как команды

```
#{\msam{\char1}} {\msam{\char39}} {\msbm{\char26}} {\msbm{\char126}}#
```

приведут к выдаче символов $\Delta \prime \ae \sim$.

\LaTeX формализует работу с коллекцией $\mathcal{A}\mathcal{M}\mathcal{S}$ как с обычными математическими шрифтами. \LaTeX располагает двумя корректирующими файлами стиля, в которых заданы команды загрузки шрифтов, и символы этих шрифтов определены надлежащими командами как математические — `amfonts` и `amssymb`. Введя их в команду описания стиля документа

```
\documentstyle[amfonts,amssymb,...]{article}
```

можно в дальнейшем для помещения в математическую часть текста требуемых символов этих шрифтов пользоваться командами, приведенными в нижеследующих таблицах.

\bumpeq	<code>\Bumpeq</code>	\cap	<code>\Cap</code>	\cup	<code>\Cup</code>
\doteq	<code>\Doteq</code>	\leftarrow	<code>\Lleftarrow</code>	\uparrow	<code>\Lsh</code>
\Rightarrow	<code>\Rrightarrow</code>	\rightarrow	<code>\Rsh</code>	\subseteq	<code>\Subset</code>
\supseteq	<code>\Supset</code>	\vdash	<code>\Vdash</code>	\Vdash	<code>\Vvdash</code>
\angle	<code>\angle</code>	\approx	<code>\approxeq</code>	ε	<code>\backepsilon</code>
\backprime	<code>\backprime</code>	\simeq	<code>\backsimeq</code>	\backsimeq	<code>\backsimeq</code>
$\bar{\wedge}$	<code>\barwedge</code>	\because	<code>\because</code>	\beth	<code>\beth</code>
\between	<code>\between</code>	\bigstar	<code>\bigstar</code>	\boxdot	<code>\boxdot</code>
\blacklozenge	<code>\blacklozenge</code>	\blacksquare	<code>\blacksquare</code>	\blacktriangle	<code>\blacktriangle</code>
\boxminus	<code>\boxminus</code>	\boxplus	<code>\boxplus</code>	\boxtimes	<code>\boxtimes</code>
\bumpeq	<code>\bumpeq</code>	\cdot	<code>\centerdot</code>	\checkmark	<code>\checkmark</code>
\circeq	<code>\circeq</code>	\textcircled{R}	<code>\circledR</code>	\textcircled{S}	<code>\circledS</code>
\circledast	<code>\circledast</code>	\complement	<code>\complement</code>	\curlyeqprec	<code>\curlyeqprec</code>
\curlyeqsucc	<code>\curlyeqsucc</code>	\curlyvee	<code>\curlyvee</code>	\curlywedge	<code>\curlywedge</code>
\daleth	<code>\daleth</code>	\digamma	<code>\digamma</code>	\doteqdot	<code>\doteqdot</code>
\dotplus	<code>\dotplus</code>	\doublecap	<code>\doublecap</code>	\doublecup	<code>\doublecup</code>
\eqcirc	<code>\eqcirc</code>	\eqslantgtr	<code>\eqslantgtr</code>	\eqslantless	<code>\eqslantless</code>
\fallingdotseq	<code>\fallingdotseq</code>	\geqq	<code>\geqq</code>	\geqslant	<code>\geqslant</code>
\ggg	<code>\ggg</code>	\gimel	<code>\gimel</code>	\gneq	<code>\gneq</code>
\gneqq	<code>\gneqq</code>	\gnsim	<code>\gnsim</code>	\gtrapprox	<code>\gtrapprox</code>
\gtrdot	<code>\gtrdot</code>	\gtreqless	<code>\gtreqless</code>	\gtreqqless	<code>\gtreqqless</code>
\gtrless	<code>\gtrless</code>	\gnapprox	<code>\gnapprox</code>	\gtrsim	<code>\gtrsim</code>
\gvertneqq	<code>\gvertneqq</code>	\hbar	<code>\hbar</code>	\hslash	<code>\hslash</code>
\intercal	<code>\intercal</code>	\leqq	<code>\leqq</code>	\leqslant	<code>\leqslant</code>
\lessapprox	<code>\lessapprox</code>	\lessdot	<code>\lessdot</code>	\lesseqgtr	<code>\lesseqgtr</code>
\lesseqqgtr	<code>\lesseqqgtr</code>	\lessgtr	<code>\lessgtr</code>	\lesssim	<code>\lesssim</code>
\llcorner	<code>\llcorner</code>	\lll	<code>\lll</code>	\lnapprox	<code>\lnapprox</code>
\lneq	<code>\lneq</code>	\lneqq	<code>\lneqq</code>	\lnsim	<code>\lnsim</code>
\lozenge	<code>\lozenge</code>	\lrcorner	<code>\lrcorner</code>	\ltimes	<code>\ltimes</code>
\lvertneqq	<code>\lvertneqq</code>	\maltese	<code>\maltese</code>	\mho	<code>\mho</code>
\multimap	<code>\multimap</code>	\nVDash	<code>\nVDash</code>	\nVDash	<code>\nVDash</code>
\nexists	<code>\nexists</code>	\ngeq	<code>\ngeq</code>	\ngeqq	<code>\ngeqq</code>
\ngeqslant	<code>\ngeqslant</code>	\ngtr	<code>\ngtr</code>	\nleftarrow	<code>\nleftarrow</code>
\nleq	<code>\nleq</code>	\nleqq	<code>\nleqq</code>	\nleqslant	<code>\nleqslant</code>
\nless	<code>\nless</code>	\nmid	<code>\nmid</code>	\nparallel	<code>\nparallel</code>
\nprec	<code>\nprec</code>	\npreceq	<code>\npreceq</code>	\nrightarrow	<code>\nrightarrow</code>
\nshortmid	<code>\nshortmid</code>	\nsim	<code>\nsim</code>	\nsubseteq	<code>\nsubseteq</code>
\nsubseteqq	<code>\nsubseteqq</code>	\nsucc	<code>\nsucc</code>	\nsucceq	<code>\nsucceq</code>
\nsupseteq	<code>\nsupseteq</code>	\nsupseteqq	<code>\nsupseteqq</code>	\nVDash	<code>\nVDash</code>
\nvdash	<code>\nvdash</code>	\restriction	<code>\restriction</code>	\rtimes	<code>\rtimes</code>

Примеры символов \mathcal{AMS} -коллекции.

Операторы отношения:	\lesseqgtr \gtrless \lll \ggg \lesseqgtr \gtrless \lesseqgtr \gtrless \lesseqgtr \gtrless
Стрелки:	\circlearrowleft \circlearrowright \leftrightarrow \rightleftarrows \curvearrowleft \curvearrowright \upuparrows \rightsquigarrow \nrightarrow \Lleftarrow \Rrightarrow
Углы:	\angle \sphericalangle \sphericalangle \llcorner \ulcorner \urcorner \lrcorner \blacktriangle
Символы:	\complement \digamma \hbar \mathcal{U} \hbar ε \textcircled{R} \textcircled{S}

```

\begin{center}
Примеры символов \mathcal{AMS}-коллекции.
\par\bigskip\begin{tabular}{l}
Операторы отношения: & \& \$ \lesseqgtr \gtrless \quad \lll \ggg \quad \lesseqgtr \gtrless \quad \lesseqgtr \gtrless \quad \lesseqgtr \gtrless \\
& \& \$ \circlearrowleft \circlearrowright \quad \leftrightsquigarrow \rightleftarrows \quad \curvearrowleft \curvearrowright \quad \Lsh \Rsh \quad \Lleftarrow \Rrightarrow \\
& \& \$ \angle \sphericalangle \quad \llcorner \ulcorner \urcorner \lrcorner \quad \blacktriangle \\
Символы: & \& \$ \complement \digamma \quad \hbar \mathcal{U} \hbar \varepsilon \quad \textcircled{R} \textcircled{S} \\
\end{tabular}
\end{center}

```

4.4. Увеличение шрифтов

Приведенные здесь команды редко используются в файлах, подготовленных для \LaTeX 'а. Цель данного раздела — дать общее понятие о некоторых принципах генерации шрифтов программой **METAFONT** и техники работы с ними, реализованной в \TeX 'е.

Создавая шрифт, **METAFONT** использует множество параметров, таких, как вес основных штрихов, высоту и ширину символов и т.д., которые в итоге определяют размер шрифта. При создании шрифта того же рисунка, но иного размера, эти параметры меняются по разным законам, что обусловлено эстетическими требованиями к шрифту.

METAFONT имеет также возможность получать шрифты различных размеров, масштабируя имеющиеся в наличии шрифты. При использовании масштабированных шрифтов следует соблюдать осторожность: пропорциональное увеличение всех элементов символов шрифта несколько нарушает полиграфические стандарты. В самом деле, например, шрифт размером в 5 типографских пунктов, увеличенный до размера в 10 пунктов, в своих пропорциях отличается (и не в лучшую сторону) от 10-пунктового шрифта, построенного по всем правилам традиционной полиграфии.

TeX предполагает, что для каждого шрифта, который может быть базовым шрифтом документа (т.е. шрифтом, которым набрана подавляющая часть текста), в его распоряжении имеются масштабированные шрифты, и количество их не меньше, чем количество введенных в TeX'e параметров `\magstepi`, предопределяющих шаги увеличения. Среди примитивов TeX'a в наборе команд имеются семь шагов увеличения. Все они заданы как основание 1.2 в степени, равной значению *i*, и дают следующие коэффициенты увеличения:

Параметр	Коэффициент
<code>\magstep0</code>	1
<code>\magstephalf</code>	1.095
<code>\magstep1</code>	1.2
<code>\magstep2</code>	1.44
<code>\magstep3</code>	1.728
<code>\magstep4</code>	2.074
<code>\magstep5</code>	2.488

Команда загрузки масштабированного шрифта имеет два формата:

```
\font<имя> = <внешнее имя шрифта> scaled \magstepi
\font<имя> = <внешнее имя шрифта> at <желаемый размер>
```

В принципе в этой команде можно указать произвольный размер шрифта, однако следует иметь в виду, что возможность масштабирования шрифтов заложена не во все драйверы и, вероятно, ваш выбор ограничен набором шрифтов, предназначенных для конкретного устройства вывода. Если нет шрифта заказанного размера, то драйвером будет использован шрифт, размер которого достаточно близок к заказанному. Но не исключена возможность, что драйвер прекратит работу с выдачей сообщения об отсутствии шрифтов требуемого размера.

Для шрифтов Computer Modern имеются наборы увеличений, обеспечивающие минимальные потребности TeX'a и L^AT_EX'a.

FONT cmr10	FONT cmr10 scaled at 17.28pt	FONT cmr17
------------	------------------------------	------------

```
{ \font\aaa=cmr10 \font\aaa=cmr10 scaled \magstep3 \font\aaa=cmr17
{\aaa FONT cmr10 \qqquad
{\aaa FONT cmr10 scaled at 17.28pt}\qqquad
{\aaa FONT cmr17}
```

4.5. Новая схема выбора шрифтов

Команды работы со шрифтами, предложенные в TeX'e и версии L^AT_EX 2.09, были разработаны тогда, когда коллекция шрифтов была сравнительно небольшой и унификация приемов, основанная на какой-либо классификации, не представлялась важной.

—

В $\text{\LaTeX}/2\text{e}$ и $\text{\LaTeX} 3$ принята новая схема выбора шрифтов (New Font Selection Scheme — NFSS). Эта схема более стандартизована и удобна в использовании, особенно для разработчиков стилей.

В новой схеме выбора шрифтов реализованы следующие важные моменты:

- Команда загрузки любого шрифта может быть выдана в любом месте документа.
- Команды изменения характеристик шрифта полностью независимы и могут использоваться в любом порядке.

Новая схема выбора шрифтов принципиально изменяет работу со шрифтами на стадии подготовки конфигурации системы для новой компиляции.

С точки зрения пользователя NFSS полностью совместима с первоначальной схемой, и лишь снимает некоторые ограничения. Например, в версии $\text{\LaTeX} 2.09$ при старой схеме выбора шрифтов надо было следить за тем, чтобы тип шрифта декларировался после задания размера, т.к. любая команда изменения размера шрифта восстанавливала прямой шрифт. Новая схема выбора шрифтов позволяет выдавать команды работы со шрифтами в любом порядке.

Мы даем краткие сведения о работе с NFSS, важные именно для пользователя.

Для активизации в $\text{\LaTeX} 3$ или $\text{\LaTeX}/2\text{e}$ работы схемы, принятой в версии $\text{\LaTeX} 2.09$, в список корректирующих стилей команды `\documentstyle` следует включить `oldlfont`.

4.5.1. Команды описания шрифта

Новая схема оперирует следующими характеристиками шрифтов:

- family** *семейство* — набор шрифтов, имеющих общие характеристики в рисунке символа, такие, например, как высота отдельных групп символов или особенное начертание некоторых деталей.
Семейство включает в себя шрифты различных размеров, форм, насыщенности и ширины.
- series** *серия* определяется комбинацией двух взаимосвязанных характеристик, насыщенности и ширины.
По насыщенности наиболее часто употребляемые шрифты разделяются на светлые, нормальные и жирные, по ширине — на узкие, нормальные и широкие. Существуют и другие варианты, например, полужирный или сверхжирный шрифты;
- shape** *форма* шрифта может быть одной из следующих: прямая, курсив, капитель, наклонная, прямой курсив;
- size** *размер* шрифта задается в типографских пунктах.

Для описания характеристик шрифтов введены следующие команды:

▷ `\family{<имя>}`

Имена семейств, принятые в новой схеме выбора шрифтов, совпадают с приведенными в разделе 4.

Примеры названий семейств:

`cmr` — Computer Modern Roman

`cmtt` — Computer Modern Typewriter

`cmsy` — Math symbols

`euf` — Euler Fraktur

`msa` — AMS symbol font A

▷ `\series{<код>}`

Код серии представляет собой комбинацию двух взаимосвязанных характеристик: насыщенности и ширины.

Для построения кода в команде `\series` нужно объединить пару соответствующих аббревиатур из таблицы 1. Нормальная ширина (значение `m`) принята по умолчанию и в название кода не вводится.

Таблица 1. Насыщенность и ширина шрифтов

Насыщенность	Ширина		
Ultra Light	<code>ul</code>	Ultra Condensed	<code>uc</code>
Extra Light	<code>el</code>	Extra Condensed	<code>ec</code>
Light	<code>l</code>	Condensed	<code>c</code>
Semilight	<code>sl</code>	Semicondensed	<code>sc</code>
Medium (normal)	<code>m</code>	Medium	<code>m</code>
Semibold	<code>sb</code>	Semiexpanded	<code>sx</code>
Bold	<code>b</code>	Expanded	<code>x</code>
Extra Bold	<code>eb</code>	Extra Expanded	<code>ex</code>
Ultra Bold	<code>ub</code>	Ultra Expanded	<code>ux</code>

Серии, присутствующие в семействе Computer Modern Roman:

`m` — нормальная(средняя) насыщенность и нормальная ширина;

`b` — жирная насыщенность и нормальная ширина;

`bx` — жирная насыщенность и полурасширенная ширина.

Серии, присутствующие в семействе Computer Modern Sans Serif:

`m` — нормальная(средняя) насыщенность и нормальная ширина;

`sbx` — полужирная насыщенность и полурасширенная ширина;
`bx` — жирная насыщенность и полурасширенная ширина.

▷ `\shape{<код>}`

Код формы шрифта может быть одним из следующих:

`n` — прямая; `sc` — капитель;
`it` — курсив; `sl` — наклонная;
`u` — прямой курсив.

▷ `\size{<число1>}{<число2>}`

Поскольку размер текущего шрифта и расстояние между двумя соседними строками сильно связаны, команда определения размера имеет два обязательных параметра.

Первый из них — это размер шрифта в типографских пунктах (без указания единиц измерения), второй — расстояние между базовыми линиями соседних строк с указанием единиц измерения.

В таблице 2 приведены параметры команды `\size`, реализованные в стандартных макрокомандах \LaTeX 'а, для различных размеров нормального шрифта документа. Собственно размер шрифта указан над косой чертой, а под чертой приведено расстояние между строками в том виде, в каком они должны появляться в команде `\size`, т.е. с указанием единиц размерности.

Таблица 2. Стандартные размеры шрифтов в \LaTeX 'е

Команда \LaTeX 'а	Размер базового шрифта в документе			
	10pt	11pt	12pt	14pt
<code>\tiny</code>	5/6pt	6/7pt	6/7pt	7/8.4pt
<code>\scriptsize</code>	7/8pt	8/9.5pt	8/9.5pt	9/11pt
<code>\footnotesize</code>	8/9.5pt	9/11pt	10/12pt	11/13.6pt
<code>\small</code>	9/11pt	10/12pt	11/13.6pt	12/14.5pt
<code>\normalsize</code>	10/12pt	11/13.6pt	12/14.5pt	14/18pt
<code>\large</code>	12/14pt	12/14pt	14/18pt	17/20pt
<code>\Large</code>	14/18pt	14/18pt	17/22pt	20/25pt
<code>\LARGE</code>	17/22pt	17/22pt	20/25pt	25/30pt
<code>\huge</code>	20/25pt	20/25pt	25/30pt	—
<code>\Huge</code>	25/30pt	25/30pt	—	—

Классификация наиболее часто используемых текстовых шрифтов Computer Modern размером в 10pt приведена в таблице 3.

Таблица 3. Классификация шрифтов Computer Modern размером в 10 пунктов

family	series	shape(s)	Имена шрифтовых файлов
<i>Гарнитура CM roman</i>			
cmr	m	n, it, sl, sc, u	cmr10, cmti10 cmsl10, cmcsc10, cmu10
cmr	bx	n, it, sl	cmbx10, cmbxti10, cmbxsl10
cmr	b	n	cmb10
<i>Гарнитура CM sans serif</i>			
cmss	m	n, it	cmss10, cmssi10
cmss	bx	n	cmssbx10
cmss	sbc	n	cmssdc10
<i>Гарнитура CM typewriter</i>			
cmtt	m	n, it, sl, sc	cmtt10, cmitt10, cmsltt10, cmtcsc10

4.5.2. Команды выбора шрифта

Задание всех характеристик шрифта выполняется посредством команд, описанных в предыдущем подразделе:

```
\family \shape \series \size
```

Для изменения отдельной характеристики шрифта следует воспользоваться одной командой, соответствующей этой характеристике.

Выбор шрифта с установленными характеристиками делается командой

▷ `\selectfont`

Нижеследующий пример построен так— команды задания шрифта приведены так, как они выглядят в исходном тексте, а строки текста дана уже в сформированном этими командами виде.

```
\family{cmr}\series{m}\shape{n}\size{14}{18pt}\selectfont
Что посеешь, то пожнешь. Good proverb!
\size{12}{14pt}\selectfont
Что посеешь, то пожнешь. Good proverb!
\series{m}\shape{it}\selectfont
Что посеешь, то пожнешь. Good proverb!
\shape{sl}\selectfont
Что посеешь, то пожнешь. Good proverb!
\shape{sc}\selectfont
ЧТО ПОСЕЕШЬ, ТО ПОЖНЕШЬ. GOOD PROVERB!
\shape{u}\selectfont
Что посеешь, то пожнешь. Good proverb!
\series{bx}\shape{it}\selectfont
Что посеешь, то пожнешь. Good proverb!
```

$\series{b}\shape{n}\selectfont$
Что посеешь, то пожнешь. Good proverb!
 $\family{cmss}\series{m}\shape{n}\selectfont$
 Что посеешь, то пожнешь. Good proverb!
 $\series{m}\shape{it}\selectfont$
Что посеешь, то пожнешь. Good proverb!
 $\series{bx}\shape{n}\selectfont$
Что посеешь, то пожнешь. Good proverb!
 $\family{cmss}\series{sbc}\shape{n}\selectfont$
Что посеешь, то пожнешь. Good proverb!
 $\family{cmtt}\series{m}\shape{n}$
Что посеешь, то пожнешь. Good proverb!
 $\shape{it}\selectfont$
Что посеешь, то пожнешь. Good proverb!
 $\shape{sl}\selectfont$
Что посеешь, то пожнешь. Good proverb!
 $\shape{sc}\selectfont$
 CORRESPONDING CYRILLIC FONT DOESN'T EXISTS.
 $\family{cmsy}\shape{n}\selectfont$
 $\% \symbol{3}\quad\symbol{32} \quad\quad\symbol{70}\quad\symbol{115}\quad\symbol{116}$

* ← \mathcal{F} \int \sqcup

Термины и определения

Настоящий список содержит глоссарий терминов и определений, которые используются в обеих частях данного руководства.

- **Визуализатор**
Специальный драйвер для просмотра сформатированного документа на экране дисплея.
- **Гарнитура**
Набор шрифтов, сохраняющих стилевые признаки основного рисунка, но отличающиеся по начертаниям и размерам. Изменение рисунка шрифта в пределах одной гарнитуры могут дать начертания, различающиеся
 - по наклону — прямое, наклонное вправо или влево;
 - по плотности — узкое, нормальное, широкое;
 - по насыщенности — светлое, жирное.Гарнитура (как и кегль) есть классическая полиграфическая характеристика шрифта.
- **Глиф**
Графическое представление символа. Например, **a**, **a** и **a** есть три различных глифа для одного и того же символа. термин образован сокращением английского слова GLYptograPHy, означающего "резьба на драгоценных камнях".
- **Драйверы**
Программы, обеспечивающие вывод создаваемых Т_EX'ом и Л_AT_EX'ом не зависящих от типа внешнего устройства файлов на конкретное устройство вывода.
- **Кегль**
Размер шрифта в типографских пунктах, включающий высоту буквы (очка) и заплечики (свободное пространство над и под очком). Кегль относится к классическим полиграфическим характеристикам шрифта.
- **Лигатуры**
Типографское представление двух или трех символов в виде одного "иероглифа", облегчающее их зрительное восприятие. Например, символы "ffl" или "---" Т_EX выдаст как "ffi" и "—".
- **Макрокоманда**
Произвольная команда, составленная из примитивов и(или) других макрокоманд. Макрокоманда может иметь параметры.

- **METAFONT**

Программа-спутник процессора текстов \TeX , используемая для генерации новых шрифтов или существующих шрифтов с новым увеличением. Программа **METAFONT** читает файл определения шрифта, фактические параметры устройства, увеличение и т.д., чтобы создать *два* выходных файла на каждый входной файл определения шрифта. Входной файл должен быть файлом типа `.mf`, а два выходных файла есть соответственно `tmf`- и `gf`-файлы. Если вы хотите сделать шрифт пяти различных увеличений, вы должны пять раз запустить на счет программу **METAFONT** с почти одинаковым `mf`-файлом. Или если вы желаете создать шрифт для трех различных устройств (матричное печатающее устройство, лазерное устройство печати, дисплей) вы опять-таки должны запустить программу **METAFONT** три раза с соответствующим определением устройства. *Все три раз будет выдан один и тот же `tmf`-файл. Однако `gf`-файлы будут различными.*

- **Метрический файл шрифта**

Аппаратно-независимый файл, содержащий набор чисел, определяющий для каждого глифа необходимое ему пространство на странице: размер бокса, заключающего в себя глиф и некоторые граничные условия, например относительные расстояния между соседними глифами. Тип файла есть `tfm`.

- **New Font Selection Scheme — NFSS**

Новая схема выбора шрифтов (New Font Selection Scheme — NFSS) введена в последних версиях \LaTeX 2.09, \LaTeX 3. В NFSS реализованы следующие важные моменты:

- Команды изменения характеристик шрифта полностью независимы и могут использоваться в любом порядке.
- Команда загрузки любого шрифта может быть выдана в любом месте документа.

- **PAW**

Система интерактивного анализа данных, интегрирующая в себе различные инструменты управления и представления данных. Графические пакеты включают возможность создания файлов в формате `POSTSCRIPT`.

- **Пиксель**

Независимый элемент растрового представления графической информации, который способно напечатать устройство. Разрешение устройства вывода определяется в точках(пикселях) на дюйм, `Dot per inch`, `DPI`, отдельно по горизонтали и вертикали. Для механических устройств типично разрешение 240DPI по горизонтали и 180DPI по вертикали. Для лазерных устройств типичное разрешение есть 300DPI. Обычно указывается разрешение только по горизонтали.

- **Примитивы**

Около трех сотен встроенных в программу \TeX команд, неразложимых на более

—
простые функциональные компоненты.

- **POSTSCRIPT**
Язык программирования для устройств печати (разработка фирмы Adobe), используемый на многих лазерных устройствах.
- **Разрешение (характеристика шрифта)**
Разрешение как характеристика шрифта определяется разрешением устройства вывода, для которого этот шрифт предназначен. Измеряется в точках(пикселях) на дюйм и обычно указывается в имени растрового файла в качестве его расширения.
- **Растривание символа**
METAFONT имеет свою внутреннюю графическую решетку или растр, состоящий из прямоугольных ячеек, внутри которых можно поместить только один пиксель. Растрезация символа есть создание пиксельной картинке его путем указания **METAFONT**'у, какие пиксели растра должны быть "черными", а какие "белыми". По существу это есть преобразование формы в двоичные комбинации.
- **Растровый шрифтовый файл**
Файл, содержащий битовое представление шрифта для конкретного устройства вывода. Очевидно, что растровое представление любого глифа в шрифте явно зависит от устройства печати и увеличения, и каждый шрифт должен храниться в отдельном файле.
- **Стиль документа**
Совокупность макрокоманд, задающих для **Л^AT_EX**'а тип создаваемого документа. Стиль бывает базовым и корректирующим. Базовый стиль определяет стандарт документа, распознаваемый **Л^AT_EX**'ом: книга, статья, письмо, доклад, препринт. Корректирующий стиль вносит в базовый поправки, дополняющие его. Например, указание корректирующего стиля "russian" при любом базовом стиле приведет к тому, что переносы в словах русских фрагментов документа будут сделаны по правилам русского языка.
- **Типографский пункт**
Основная единица размерности, используемая в типографском деле. Один типографский пункт равен 0,3759 мм.
- **Типографский размер шрифта (design size)**
Размер, выбранный при проектировании шрифта данного начертания. Измеряется в типографских пунктах. Программа **METAFONT**, может создавать посредством механического увеличения аналогичные шрифты других, отличных от типографского, размеров.

- **Формат**
Заданный набор макрокоманд, откомпилированных и записанных в файл типа FMT, использование которого позволяет избежать траты времени на повторную компиляцию.
- **Шрифт (фонт)**
Набор глифов, соответствующих набору символов.

Алфавитный указатель

abbrv (библ. стиль), 4
alpha (библ. стиль), 4
amsmath, 19
amssymb, 19

\bBitMap, 12
\bibliography, 4
\bibliographystyle, 3, 4
BibTeX (программа), 4, 5, 9
\BitMap, 12
BitMap (корр. стиль), 11
\BitMapDirectory, 14
\BitMapResolution, 14

\cBitMap, 12
\char, 18
\cite, 3, 4

\documentstyle, 1, 11

\family, 25
\font, 17
\footnotesize, 26

\HideDisplacementBoxes, 13
\hSlide, 13
\Huge, 26
\huge, 26

\index, 1, 2

\LARGE, 26
\Large, 26
\large, 26

\magstep, 23
makeidx (корр. стиль), 1
MakeIndex (программа), 1
\makeindex, 1

\newfont, 17
\nocite, 3, 4
\normalsize, 26

oldfont (корр. стиль), 24

rcxsize (программа), 11
plain (библ. стиль), 3

\printindex, 1

\scriptsize, 26
\selectfont, 27
\series, 25
\shape, 26
\ShowDisplacementBoxes, 13
\size, 26
\small, 26
\symbol, 18

\tBitMap, 12
\thebibliography, 3
\tiny, 26
\TrimBottom, 13
\TrimBoundingBox, 13
\TrimLeft, 13
\TrimRight, 13
\TrimTop, 13

unsrc (библ. стиль), 4

\vSlide, 13

Список литературы

- [1] Knuth D.E.: *The T_EXBook*. Addison-Wesley, Reading, Massachusetts, 1984.
- [2] Lamport L.: *Л_AT_EX: A Document Preparation System*. Addison-Wesley, Reading, Massachusetts, 1986.
- [3] v.Bechtosheim Stephan, Old Oak Drike West Lafayette, USA. *T_EX in Practice*, 1989.
- [4] Sauver J.E.St., Oregon Eugene, USA: *Using T_EX on the VAX to Typeset Documents: a Primer*, 1990.
- [5] Глonti Н.Л., Грицаенко И.А. и др. *T_EX в ИФВЭ 5. Шрифты и работа с ними*. Препринт ИФВЭ 92–127, Протвино, 1992.
- [6] Глonti Н.Л., Грицаенко И.А. и др. *Многоязычный Л_AT_EX*. Препринт ИФВЭ 94–57, Протвино, 1994.
- [7] Goossens M., Mittelbach F. and Samarin A.: *The Л_AT_EX Companion*, Addison-Wesley, Reading, Massachusetts, 1994.
- [8] Diller A.: *Л_AT_EX Line by Line*, John Wiley & Sons, Chichester, West Sussex, 1993.
- [9] Львовский С.М.: *Набор и верстка в пакете Л_AT_EX*, Космоинформ, Москва, 1994.

Содержание

1. Построение алфавитного указателя	1
2. Использование баз данных при составлении библиографии	3
2.1. Команды работы с библиографией, хранящейся в базе данных	3
2.2. Процедура получения списка литературы из библиографических баз данных	4
2.3. Формат файла библиографических данных	4
2.3.1. Типы записей	5
2.3.2. Преамбула	8
2.3.3. Символьные последовательности	8
2.3.4. Ввод информации об авторах и редакторах	9
2.4. Несколько замечаний	9
3. Вставка иллюстраций	11
3.1. Вставка картинок примитивом <code>\special</code>	11
3.2. Вставка растровых картинок в формате РСХ средствами ЛАТ _E X'a	11
3.2.1. Задание метрической информации	11
3.2.2. Команды вставки иллюстраций	12
3.2.3. Команды работы с рамкой	12
4. Шрифты	15
4.1. Классификация шрифтов	15
4.2. Простейшие команды выбора шрифта и отдельного символа	17
4.3. Как использовать в математической моде шрифты коллекции АМ _S -Т _E X	19
4.4. Увеличение шрифтов	22
4.5. Новая схема выбора шрифтов	23
4.5.1. Команды описания шрифта	24
4.5.2. Команды выбора шрифта	27
Термины и определения	29
Алфавитный указатель	33
Список литературы	34