

ПРАВИЛА ОФОРМЛЕНИЯ ДОКЛАДОВ СЕМИНАРА «O & ML»*

М. И. Григорьев
m_grigoriev@list.ru

Н. В. Чашников
nik239@list.ru

9 февраля 2022 г.

1. Структура файлов с текстом доклада

Текст доклада оформляется в виде следующего набора файлов:

- стилевой файл `oml.sty` (следует использовать последнюю версию с сайта семинара: www.apmath.spbu.ru/oml/sty/oml.sty),
- основной `tex`-файл (см. шаблон на сайта семинара: <http://www.apmath.spbu.ru/oml/tex/sample.tex>),
- дополнительные файлы с рисунками и таблицами, включаемые в основной файл.

При этом имя основного файла должно быть кратким, на английском языке, отражающим суть названия доклада. Например, для файла доклада «Теорема Куна–Таккера в дифференциальной форме» можно использовать имя `KuhnTucker.tex`.

2. Оформление текста доклада

Текст доклада разбивается на разделы при помощи команды `\section{}`. При необходимости разделы можно разбить на подразделы, используя команду `\subsection{}`. Если требуется дать разделам (подразделам) заголовки, то кроме указания заголовка в параметре команды `\section`, следует также включить опцию `headers` при подключении пакета `oml`:

```
\usepackage[headers]{oml}.
```

*Семинар по оптимизации, машинному обучению и искусственному интеллекту «O&ML»
<http://www.apmath.spbu.ru/oml/>

Теоремы, леммы, определения и другие подобные элементы задаются при помощи соответствующих окружений, полный список которых можно найти в файле `oml.sty`. Для именованных теорем можно указать название в качестве необязательного аргумента окружения:

```
\begin{theorem*}[Ферма] Уравнение  $x^n+y^n=z^n$  неразрешимо
в натуральных числах при  $n \geq 3$ .
\end{theorem*}
```

Доказательство теорем оформляется при помощи окружения `proof`.

Список литературы приводится в конце доклада. Примеры оформления ссылок, в частности, ссылок на доклады семинара, приведены в шаблоне (<http://www.apmath.spbu.ru/oml/tex/sample.tex>).

Следует правильно употреблять знаки дефиса, короткого и длинного тире. Символ «-» означает дефис в тексте и знак минуса в формулах. Короткое тире (команда `--`) используется при задании диапазонов, например, для диапазонов страниц в списке литературы. Длинное тире (команда `---`) употребляется как тире в тексте.

В тексте доклада необходимо придерживаться правил русского языка, в том числе в части использования буквы «ё».

3. Метки и ссылки

Если требуется сослаться на формулу, теорему, раздел или рисунок доклада, то нужный элемент помечается при помощи команды `\label{имя}`. Имя метки должно состоять из префикса, соответствующего типу элемента (`eq` для формулы, `th` для теоремы, `fig` для рисунка), и собственно имени элемента. В качестве имени элемента можно использовать либо краткое символическое название, либо порядковый номер этого элемента в тексте. Например, если формула преобразования Фурье является первой формулой в тексте, то её можно пометить меткой `\label{eq:fourier}` или `\label{eq1}`. Для ссылок на формулы используется команда `\eqref`, а для остальных ссылок — команда `\ref`.

Пункты из списка литературы помечаются метками `bibitem{имя}`, при этом имя должно начинаться с префикса `bib`. Для ссылок на литературу используется команда `\cite`, в необязательном аргументе которой можно указать дополнительную информацию (номер страницы и т. п.):

```
\cite[c.~42]{bib:Ferma}.
```

4. Формулы

4.1 Формулы внутри строки

Формулы внутри строки, ограниченные символами `$`, должны быть по возможности короткими. Не нужно включать знаки препинания внутрь формул: вместо `$a, \, b$` лучше писать `a, b`. Также следует избегать разрывов строк внутри таких формул. Если без разрыва строки не обойтись, и разрыв приходится на знак операции, надо добавить перед этим знаком команду `\hyphmath`: например, `a+b\hyphmath=c`. Тогда при переносе знак операции будет повторен в начале строки.

4.2 Формулы на отдельной строке

Формулы, расположенные на отдельной строке (*выключные* формулы) оформляются при помощи окружения `equation*` или ограничиваются символами `$$`. Если формула должна быть пронумерована, используется окружение `equation`.

4.3 Группа формул

Если несколько выключных формул идут подряд в тексте, они должны быть объединены в одном окружении. В случае, если формулы разнородные, можно использовать окружение `gather*` для выравнивания по центру:

```
\begin{gather*}
  f(x)\to\min, \\
  a(x)\ge 0.
\end{gather*}
```

$$f(x) \rightarrow \min,$$

$$a(x) \geq 0.$$

Если формулы однородные (например, представляют собой систему уравнений), следует использовать окружение `align*`, указывая знаком `&` символ, по которому формулы должны быть выровнены:

```
\begin{align*}
  a_1\,x + b_1\,y &= c_1, \\
  a_2\,x + b_2\,y &= c_2.
\end{align*}
```

$$a_1 x + b_1 y = c_1,$$

$$a_2 x + b_2 y = c_2.$$

Для однородных формул, состоящих из нескольких столбцов, подходит окружение `alignat*`. Параметр окружения задаёт количество столбцов, знаками `&` указываются символы, по которым производится выравнивание. Столбцы также отделяются друг от друга знаком `&`, при этом расстояние между столбцами надо задать явно в одной из строчек:

```

\begin{alignat*}{2}
a_i(x) &\geq 0, \quad \text{\quad} & i \in M_1, \\
a_i(x) &= 0, \quad & i \in M_2.
\end{alignat*}

```

Если требуется более сложное выравнивание, например, часть формул надо выровнять по знаку равенства и центрировать с остальными формулами, то можно комбинировать приведённые выше окружения, используя варианты с суффиксом `ed` (`gathered`, `aligned` и `alignedat`) для вложенных формул:

```

\begin{gather*}
f(x) \to \min, \\
\begin{alignedat}{2}
a_i(x) &\geq 0, \quad \text{\quad} & i \in M_1, \\
a_i(x) &= 0, \quad & i \in M_2.
\end{alignedat}
\end{gather*}

```

Для того, чтобы пронумеровать внутренние формулы по отдельности, применяются варианты окружений без звёздочек. При этом можно вставить команду `\notag` перед `\\` для тех строчек, где нумерация не нужна. Если же требуется пронумеровать всю систему целиком, то надо заключить её в окружение `equation`, заменив окружения на их варианты с суффиксом `ed`.

4.4 Многострочные формулы

Предположим, что длинная цепочка равенств или неравенств не помещается на одной строке. В таком случае предпочтительно разбить формулу по знакам равенства (неравенства) и выровнять её, используя окружение `align*`, таким образом, чтобы знаки были один над другим:

```

\begin{align*}
f(x) &= (a+b)^2 = \\
&= a^2 + 2ab + b^2.
\end{align*}

```

Если же даже часть формулы между соседними знаками равенства не помещается в одну строку, можно разбить формулу по знаку операции. При этом на место отсутствующего операнда ставится конструкция `{}`, чтобы обеспечить правильные пробелы, а перенесённая часть формулы выравнивается со сдвигом относительно знака равенства:

$$\begin{aligned}
& \text{\code{\begin{align*}}} & f(x) = \sum_i a_i + \sum_i b_i + \\
& \text{\code{f(x)\&= \sum_i a_i + \sum_i b_i +\{\}\}} \\
& \quad \text{\code{\quad \{\} + \sum_i c_i = \}} \\
& \quad \text{\code{\&= \sum_i (a_i + b_i + c_i).}} & + \sum_i c_i = \\
& \text{\code{\end{align*}}} & = \sum_i (a_i + b_i + c_i).
\end{aligned}$$

В крайнем случае, если другие варианты не подходят, можно центрировать все строчки в цепочке равенств при помощи окружения `gather*`.

4.5 Текст внутри формул

Для вставка текста внутрь выключной формулы применяется команда `\text`. Типичный пример — окружение `cases`, используемое для формул с разбором случаев:

$$\begin{aligned}
& \text{\code{\$\$}} & \\
& \text{\code{\delta_{ij} =}} & \\
& \text{\code{\begin{cases}}} & \\
& \text{\code{1, \quad \&text{если \$i=j\$,}}}} & \delta_{ij} = \begin{cases} 1, & \text{если } i = j, \\ 0, & \text{иначе.} \end{cases} \\
& \text{\code{0, \quad \&text{иначе.}}} & \\
& \text{\code{\end{cases}}} & \\
& \text{\code{\$\$}} &
\end{aligned}$$

Часть формулы, непосредственно связанную с текстом, лучше помещать внутрь аргумента команды `\text`, как в приведённом примере.

Если требуется вставить текст на отдельную строчку внутри многострочной формулы, не нарушая выравнивания, используется команда `\intertext`:

$$\begin{aligned}
& \text{\code{\begin{align*}}} & f(x, y, z) = a(x, y, z) + b(x, y, z), \\
& \text{\code{f(x,y,z) \&= a(x,y,z)+b(x,y,z), \}} & \\
& \text{\code{\intertext{где}}} & \text{где} \\
& \text{\code{a(x,y,z) \&= x+y, \}} & a(x, y, z) = x + y, \\
& \text{\code{b(x,y,z) \&= y+z.}} & b(x, y, z) = y + z. \\
& \text{\code{\end{align*}}} &
\end{aligned}$$

4.6 Скобки

Не рекомендуется использовать команды `\left` и `\right` для задания размера скобок. Вместо этого следует задавать размер скобок явно, используя команды `\bigl`, `\Bigl`, `\biggl` и `\Biggl` для открывающихся скобок и аналогичные команды с `r` на конце для закрывающихся скобок. Внешние скобки

обычно должны иметь больший размер, чем вложенные:

$$\mathbb{f}\bigl(a(x)\bigr) \qquad f(a(x))$$

В ситуациях, когда уровней вложенных скобок так много, что внешние скобки получаются слишком большими, можно либо переписать формулы таким образом, чтобы уменьшить количество уровней, либо сделать некоторые скобки одинакового размера:

$$\mathbb{f}\bigl\langle f(a(x)), b(x) \bigr\rangle \qquad \langle f(a(x)), b(x) \rangle$$

4.7 Дроби

В некоторых случаях команда `\frac` не даёт дробь желаемого размера. Тогда можно использовать команду `\tfrac`, чтобы получить маленькую дробь, и команду `\dfrac`, чтобы получить большую.

5. Пробелы

5.1 Неразрывный пробел

Неразрывный пробел (команда `~`) употребляется в случаях, когда разрыв строки между соседними словами нежелателен. Вот основные ситуации, в которых требуется неразрывный пробел: перед однобуквенными формулами (для целых n), перед ссылками (см. `\eqref{eq:1}`), перед тире (миру --- мир).

5.2 Горизонтальные пробелы

Иногда пробелы, расставляемые \TeX 'ом в формулах, нуждаются в корректировке. Рассмотрим используемые для этого команды и типичные случаи их применения:

- отрицательный пробел (команда `\!`) — используется для уменьшения промежутка, например, между большой матрицей и следующим за ней знаком препинания;
- тонкий пробел (команда `\,`) нужен, в частности, для отделения друг от друга выражений с индексами (a_i, b_j) и отделения выражения с индексами или дроби от последующего знака препинания в выключной формуле;
- средний и толстый пробелы (команды `\:` и `\;`) могут применяться между элементами перечисления ($\{x>0, \; y>0\}$);

- большие пробелы (команды `\quad` и `\qquad`) используются для отделения формул друг от друга и отделения текста от формулы:

$$a_i(x) > 0, \qquad i \in M.$$

Для правильной расстановки пробелов следует также употреблять команду `\colon` вместо двоеточия в формулах вида

$$f \colon [0, 1] \to [0, 1]$$

и команду `\mid` вместо вертикальной черты при описании множеств:

$$\{x \in \mathbb{R} \mid x > 2\}$$

5.3 Вертикальные пробелы

Если в многострочной формуле большое число индексов, то вертикальный промежуток между строчками может оказаться слишком маленьким. Чтобы его увеличить, следует добавить к команде разрыва строки аргумент, указывающий величину дополнительного промежутка (`\smallskipamount`, `\medskipamount` или `\bigskipamount`):

```
\begin{gather*}
  a_1^2+b_1^2+a_2^2+b_2^2=\\[\smallskipamount]
  =x_1^2+y_1^2+x_2^2+y_2^2.
\end{gather*}
```

Для того, чтобы увеличить вертикальный промежуток между абзацами, применяются команды `\smallskip`, `\medskip` и `\bigskip`.

6. Рисунки

6.1 Окружение figure

Иллюстрации в текст доклада добавляются как «плавающие» объекты с помощью окружения `figure`. Желаемое расположение объекта задается необязательным параметром окружения `figure`. Значение данного параметра конструируется путем собирания в цепочку следующих символов:

h	здесь же, в том самом месте текста, где появился плавающий объект
t	наверху страницы
b	внизу страницы
p	на специальной странице, содержащей только плавающие объекты
!	выполнить указанное следом размещение, даже если это будет выглядеть не очень хорошо

Например,

```
\begin{figure}[ht]
  \fbox{CNSA}
\end{figure}
```

задаёт плавающий объект, состоящий из текста `CNSA`, окружённого рамкой, и пытается разместить его в текущем месте текста, а если это невозможно, то наверху следующей страницы. Отметим, что при подключении пакета `float` становится доступным дополнительный параметр размещения `H`, который во многих случаях даёт более ожидаемый результат, чем параметр `h`.

Для вывода подписи к рисунку используется команда `\caption`. Версия данной команды со `*` используется для подписи рисунка без номера.

```
\begin{figure}[ht]
  \centering
  \fbox{0 \& ML}
  \caption{Название данного семинара}
\end{figure}
```

6.2 Пакет PSTricks

Для создания векторных рисунков рекомендуется использовать пакет `PSTricks`. Этот пакет предоставляет набор команд для различных геометрических объектов: точек, отрезков, окружностей, графиков функций и других. Координаты точек для команд указываются вещественными числами в условных единицах и задаются в форме (x, y) . Реальный размер условных единиц определяется параметром `unit`. Значение этого параметра, как и значения других параметров, устанавливаются при помощи команды `psset`.

Команды, задающие объекты рисунка, помещаются внутрь окружения `pspicture`. У этого окружения два обязательных параметра: координаты левой нижней и правой верхней точек прямоугольника, внутри которого содержится рисунок. Окружение `pspicture` включается внутрь окружения `figure`, и при этом часто используется команда `\centering`, чтобы центрировать рисунок. Рассмотрим пример:

```
\begin{figure}[h]
  \centering
  \psset{unit=1cm}
  \begin{pspicture}(0,0)(10,5)
  \end{pspicture}
\end{figure}
```


В этом примере создаётся пустой рисунок шириной 10 сантиметров и высотой 5 сантиметров, расположенный посередине страницы. Часто размеры рисунка задаются в долях ширины страницы (`\linewidth`). Например, следующие команды определяют квадратный рисунок шириной в половину страницы:

```
\psset{unit=0.5\linewidth}
\begin{pspicture}(0,0)(1,1)
\end{pspicture}
```

Перечислим команды для основных геометрических объектов:

- `\psline(x1,y1)(x2,y2)` — отрезок, соединяющий указанные точки;
- `\pspolygon(x1,y1)(x2,y2) \dots (xn,yn)` — многоугольник;
- `\psdot(x,y)` — точка с заданными координатами;
- `\pscircle(x,y){r}` — окружность с центром (x, y) и радиусом r ;
- `\psellipse(x,y)(a,b)` — эллипс с центром (x, y) и полуосями a и b ;
- `\psbezier(x1,y1)(x2,y2)(x3,y3)(x4,y4)` — кривая Безье третьего порядка с заданными контрольными точками;
- `\uput[u](x,y){text}` — поместить заданную надпись над точкой (x, y) .

Команды `\pspolygon`, `\pscircle` и `\psellipse` имеют варианты со звёздочкой для задания заполненного многоугольника, окружности и эллипса соответственно. У команды `\uput` в квадратных скобках указывается расположение надписи относительно заданной точки. Вместо буквы `u` можно использовать `l`, `r` или `d`, чтобы поместить надпись слева, справа или снизу от точки.

У команд также есть необязательные параметры. Они указываются в квадратных скобках после имени команды в виде пар **имя=значение**, перечисленных через запятую. Те же параметры и в таком же виде могут быть указаны в команде `\psset`, но тогда они будут действовать на весь рисунок, тогда как параметры, указанные в квадратных скобках после имени команды, действуют только в пределах данной команды. Например, используя параметр `linewidth`, задающий толщину линии, можно изобразить крестик, у которого горизонтальная линия жирная:

```
\psline[linewidth=2pt](0,1)(2,1)
\psline(1,0)(1,2)
```

Если же требуется, чтобы обе линии были жирными, то лучше воспользоваться командой `\psset`:

```
\psset{linewidth=2pt}
\psline(0,1)(2,1)
\psline(1,0)(1,2)
```

Параметр `linewidth` задаёт толщину линий не только для отрезков, но также для окружностей, эллипсов и других объектов.

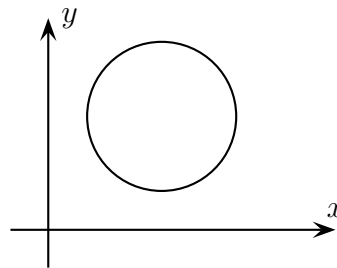
Некоторые команды имеют специфические параметры, указываемые в фигурных скобках. Например, в команде `\psline` можно задать вид концов отрезка:

```
\psline{->}(0,0)(1,1)
```

нарисует стрелку из точки $(0,0)$ в точку $(1,1)$. По умолчанию усики стрелки слишком маленькие, но их можно увеличить при помощи параметра `arrowscale`.

Приведём пример рисунка, построенного с использованием описанных выше команд:

```
\begin{figure}[h]
\centering
\psset{unit=1cm,arrowscale=2}
\begin{pspicture}(-1,-1)(4,3)
\psline{->}(-0.5,0)(3.8,0)
\psline{->}(0,-0.5)(0,2.8)
\uput[u](3.8,0){$x$}
\uput[r](0,2.8){$y$}
\pscircle(2,2){1}
\end{pspicture}
\end{figure}
```



Если рисунок достаточно большой, то целесообразно вынести его в отдельный файл с расширением `tex` и включить этот файл в основной при помощи команды `\input{имя_файла}`.

Здесь приведена лишь малая часть команд и параметров, доступных в пакете `PSTricks`. Подробная документация по этому пакету находится в подкаталоге `doc\generic\pstricks` каталога установки `MikTeX`.

7. Остальные вопросы

По всем вопросам, касающимся оформления докладов, можно обращаться к секретарю семинара.