

# МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего  
образования «Кемеровский государственный университет» Институт фундаментальных  
наук Кафедра ЮНЕСКО по информационным вычислительным технологиям

## Отчет

по учебной практике, технологической (проектно-технологической)  
практике

проект “Инструменты для оформления научных статей и  
презентаций на примере L<sup>A</sup>T<sub>E</sub>X’

### Выполнили:

студенты направления подготовки 02.03.03 Математическое обеспечение и  
администрирование информационных систем

Басалаев Дмитрий

Ф.И.О.

Оценка

Болковая Полина

Ф.И.О.

Оценка

Буданцев Артём

Ф.И.О.

Оценка

Кемерово 2021

# Содержание

<b>1</b>	<b>“Описание проекта”</b>	<b>2</b>
1.1	Актуальность, теоретическая и практическая значимость . . . . .	2
1.2	Теоретическая значимость . . . . .	2
1.3	Состав группы участников проекта . . . . .	2
1.4	Общие цель и задачи . . . . .	2
1.5	Распределение по ролям . . . . .	2
1.6	План-график работы . . . . .	3
1.7	Ссылка на репозиторий . . . . .	3
1.8	Что такое T <sub>E</sub> X и L <sup>A</sup> T <sub>E</sub> X? . . . . .	3
1.9	Используемые программные средства . . . . .	4
1.10	Что представляет собой L <sup>A</sup> T <sub>E</sub> X-документ . . . . .	4
<b>2</b>	<b>Ход работы</b>	<b>4</b>
2.1	03.02.21-11.03.21 . . . . .	4
2.2	12.03.21-26.03.21 . . . . .	4
2.3	27.03.21-15.04.21 . . . . .	5
2.4	16.04.21-28.04.21 . . . . .	7
2.5	29.04.21-14.05.21 . . . . .	11
2.6	15.05.21- . . . . .	15
<b>3</b>	<b>Выводы</b>	<b>15</b>
<b>4</b>	<b>Литература</b>	<b>15</b>

# 1 “Описание проекта”

Краткое описание: Составить презентацию и отчет о проделанной работе при помощи  $\text{\LaTeX}$ .

## 1.1 Актуальность, теоретическая и практическая значимость

**Актуальность:** Издательский пакет  $\text{\LaTeX}$  позволяет качественно оформить любой документ или презентацию, не задумываясь о её внешнем виде, а лишь сосредоточившись на изложении и структуре. С его помощью можно легко подготовить любой документ, начиная от доклада или объемного конспекта до семестровой или курсовой работы с многочисленными формулами.

## 1.2 Теоретическая значимость

- Знакомство студентов с издательским пакетом  $\text{\LaTeX}$ , описание его преимуществ и недостатков
- Получение нами умения создавать качественные pdf документы

## 1.3 Состав группы участников проекта

### Состав группы

№	ФИО	группа	Логин на github.com
1.	Басалаев Д.А.	МОА-205	FySyZe
2.	Болковая П.А.	МОА-205	ApollinariaB
3.	Буданцев А.А.	МОА-205	Anturlum

## 1.4 Общие цель и задачи

**Цель:** Составить презентацию и отчет о проделанной работе при помощи  $\text{\LaTeX}$ , задействовав как можно больше его возможностей. Познакомить студентов с его основными возможностями, обозначить некоторые преимущества по сравнению с Microsoft Word.

## 1.5 Распределение по ролям

**Басалаев Д.А.** Работа с презентациями, форматирование страницы

**Болковая П.А.** Работа с изображениями и встроенной графикой

**Буданцев А.А.** Ввод формул, различные окружения, разметка страницы.

## 1.6 План-график работы

Даты	Действия
03.02.21-11.03.21.	Изучение базы, установка необходимого софта, подготовка документации
12.03.21-26.03.21	Изучение интерфейса в $\text{\TeX}$ maker, набор простых текстов, спецсимволы
27.03.21-15.04.21	Ввод математических формул, ввод матриц, спецсимволы.
16.04.21-28.04.21	Работа с изображениями и встроенной графикой, построение графиков.
29.04.21-14.05.21	Работа с ссылками, разметка страницы, различные окружения, работа с презентациями
15.05.21-	Разработка финального продукта, подготовка отчета.

## 1.7 Ссылка на репозиторий

<https://github.com/Antur1um/Practice-LateX.git>

## 1.8 Что такое $\text{\TeX}$ и $\text{\LaTeX}$ ?

$\text{\TeX}$  — издательская система, созданная американским математиком и программистом Дональдом Кнутом (Donald E. Knuth).  $\text{\TeX}$  был разработан, преследуя две основные цели: - позволить всем создавать качественные публикации с разумными для этого усилиями.  $\text{\TeX}$  знаменит своей чрезвычайной стабильностью, работой на различных операционных системах и практически полным отсутствием ошибок. Одна из главных причин по которой  $\text{\TeX}$  выбирают для оформления научных работ заключается в том, что с его помощью можно достаточно легко вводить сложные формулы.

$\text{\LaTeX}$  — наиболее популярный набор макрорасширений (или макропакет) системы компьютерной вёрстки  $\text{\TeX}$ , который облегчает набор сложных документов. Первая версия  $\text{\LaTeX}$  была написана в 1984 году Лесли Лампортом (Leslie Lamport) и с тех пор стала доминирующим способом подготовки  $\text{\TeX}$  публикаций. Важно заметить, что ни один из макропакетов для  $\text{\TeX}$ 'а не может расширить  $\text{\TeX}$ 'овских возможностей (всё, что можно сделать в  $\text{\LaTeX}$ 'е, можно сделать и в  $\text{\TeX}$ 'е), но, благодаря различным упрощениям, использование макропакетов зачастую позволяет избежать весьма изощрённого программирования. Пакет позволяет автоматизировать многие задачи набора текста и подготовки статей, включая набор текста на нескольких языках, нумерацию разделов и формул, перекрёстные ссылки, размещение иллюстраций и таблиц на странице, ведение библиографии и др. Кроме базового набора существует множество пакетов расширения  $\text{\LaTeX}$ .

## 1.9 Используемые программные средства

1. Github
2. T<sub>E</sub>XLive
3. T<sub>E</sub>Xmaker

Для того чтобы использовать L<sup>A</sup>T<sub>E</sub>X на современном ПК под управлением Windows 10 нам понадобится загрузить и установить T<sub>E</sub>Xlive manager (это наиболее полный дистрибутив L<sup>A</sup>T<sub>E</sub>X), а также T<sub>E</sub>Xmaker (это редактор для создания T<sub>E</sub>X документов). А для сохранения документов в формате pdf нам понадобится написать пару строк в командной строке.

## 1.10 Что представляет собой L<sup>A</sup>T<sub>E</sub>X документ

L<sup>A</sup>T<sub>E</sub>X документ состоит из двух частей: файл с расширением .tex в котором содержится обычный текст и команды L<sup>A</sup>T<sub>E</sub>X (входной файл) и собственно скомпилированный pdf файл (выходной файл). Для того чтобы получить pdf файл из .tex файла нам необходимо зайти в командную строку, затем при помощи команды "cd" перейти в директорию в которой лежит .tex файл затем написать команду "pdflatex" и название файла с указанием расширения (.tex). (например: pdflatex FinalReport.tex).

## 2 Ход работы

### 2.1 03.02.21-11.03.21

Загрузили T<sub>E</sub>Xlive manager и T<sub>E</sub>Xmaker. Ознакомились с интерфейсом, синтаксисом набора команд и структурой документа. Подготовили документацию по проекту.

### 2.2 12.03.21-26.03.21

Изучили набор команд для написания спец. символов и изменения шрифта ({**жирный**, *Курсив*, крошечный **Огромный** } \$ € и др.) Решили составить таблицу, содержащую наиболее часто используемые команды, но вскоре отказались от этой идеи ибо в T<sub>E</sub>Xmaker присутствуют автоматические подсказки, а также многие действия вынесены на кнопки интерфейса.

## Основные команды для работы с текстом

Обработка внешнего вида и размера текста

Команда	Действие	Обратная(если имеется)
<code>\bfseries</code>	Полужирный шрифт	<code>\mdseries</code>
<code>\slshape</code>	Курсив	<code>\upshape</code>
<code>\Huge</code>	Увеличивает размер(макс)	
<code>\huge</code>	Чуть меньше	
<code>\LARGE</code>	Меньше	
<code>\Large</code>	Ещё меньше	
<code>\large</code>	Самый маленький среди больших	
<code>\normalsize</code>	Размер по умолчанию	
<code>\small</code>	Меньше	
<code>\footnotesize</code>	Ещё меньше	
<code>\scriptsize</code>	Маленький	
<code>\tiny</code>	Крошечный	

## Различные спец. символы

Команда	Символ	Замечания
<code>\{</code> <code>\}</code>	{ }	
<code>\\$</code> или <code>\textdollar</code>	\$	
<code>\texteuro</code>	€	Только в пакете { lmodern, texcomp }
<code>\pounds</code>	£	Только в пакете { lmodern, texcomp }
<code>\textcent</code>	¢	Только в пакете { lmodern, texcomp }
<code>\textwon</code>	₩	Только в пакете { lmodern, texcomp }
<code>\textyen</code>	¥	Только в пакете { lmodern, texcomp }

Рис. 1: Та самая недоделанная таблица

```
\begin{tabular}{|l|l|l|l|}
\hline
\bfseries \large Команда & {\bfseries \large Действие} & {\bfseries \large Обратная(если имеется)} & \\
\hline
\bfseries \bfseries & {\bfseries \large Полужирный шрифт} & \mdseries & \\
\bfseries \slshape & {\bfseries \large Курсив} & \upshape & \\
\bfseries \Huge & {\bfseries \large Увеличивает размер(макс)} & & \\
\bfseries \huge & {\bfseries \large Чуть меньше} & & \\
\bfseries \LARGE & {\bfseries \large Меньше} & & \\
\bfseries \Large & {\bfseries \large Ещё меньше} & & \\
\bfseries \large & {\bfseries \large Самый маленький среди больших} & & \\
\bfseries \normalsize & {\bfseries \large Размер по умолчанию} & & \\
\bfseries \small & {\bfseries \large Меньше} & & \\
\bfseries \footnotesize & {\bfseries \large Ещё меньше} & & \\
\bfseries \scriptsize & {\bfseries \large Маленький} & & \\
\bfseries \tiny & {\bfseries \large Крошечный} & & \\
\hline
\end{tabular}
\bigskip

\begin{center}
\bfseries {\bfseries \large Различные спец. символы} \mdseries
\end{center}

\begin{center}
\begin{tabular}{|l|l|l|l|}
\hline
\bfseries \bfseries & {\bfseries \large Символ} & {\bfseries \large Замечания} & \\
\hline
\textbackslash \{ & { & & \\
\textbackslash \} & } & & \\
\textbackslash \$ & \$ & & \\
\textbackslash \texteuro & \texteuro & Только в пакете { lmodern, texcomp } & \\
\textbackslash \pounds & \pounds & Только в пакете { lmodern, texcomp } & \\
\textbackslash \textcent & \textcent & Только в пакете { lmodern, texcomp } & \\
\textbackslash \textwon & \textwon & Только в пакете { lmodern, texcomp } & \\
\textbackslash \textyen & \textyen & Только в пакете { lmodern, texcomp } & \\
\textbackslash \# & \# & & \\
\textbackslash \% & \% & & \\
\textbackslash \$ & \$ & & \\
\textbackslash P & P & & \\
\textbackslash \textbackslash & \textbackslash & & \\
\hline
\end{tabular}
\end{center}
```

Рис. 2: Код таблицы

## 2.3 27.03.21-15.04.21

Итак, мы приступили к вводу математических выражений и формул. Желая начать с чего-то простого мы решили переписать школьную таблицу производных и интегралов.

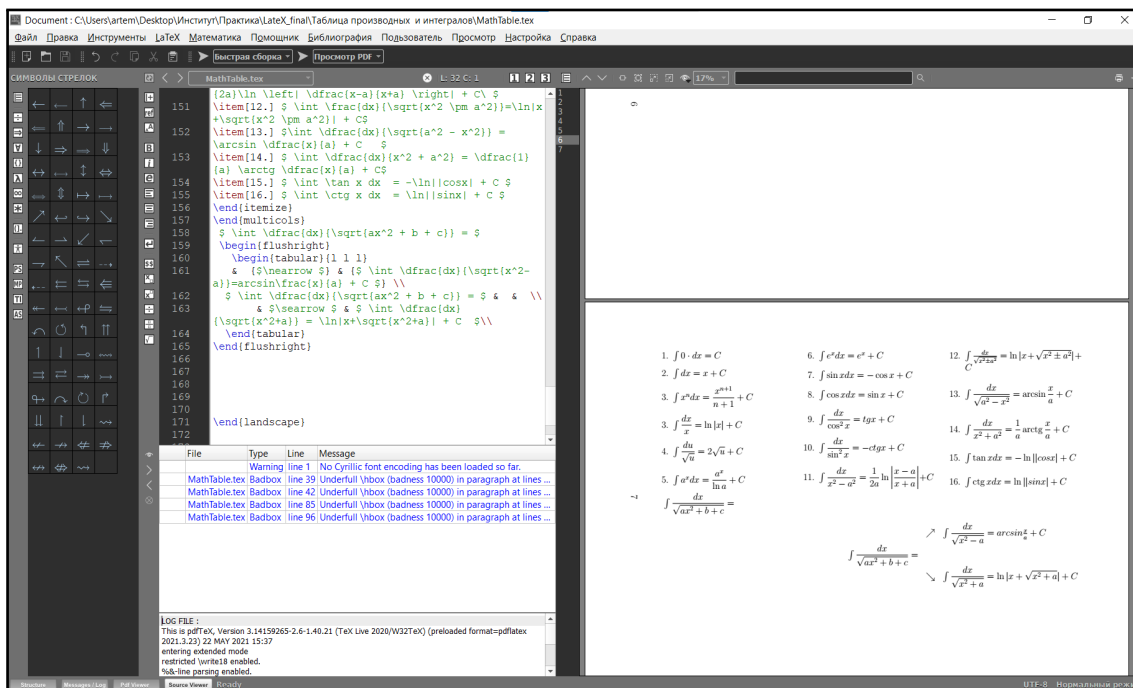


Рис. 3: Уже на этом этапе можно понять насколько в  $\text{\LaTeX}$  проще и быстрее вводить математические формулы

Итак, быстро убедившись что ввод сложных математических формул не представляет трудностей мы приступили к вводу матриц и других крупных объектов.

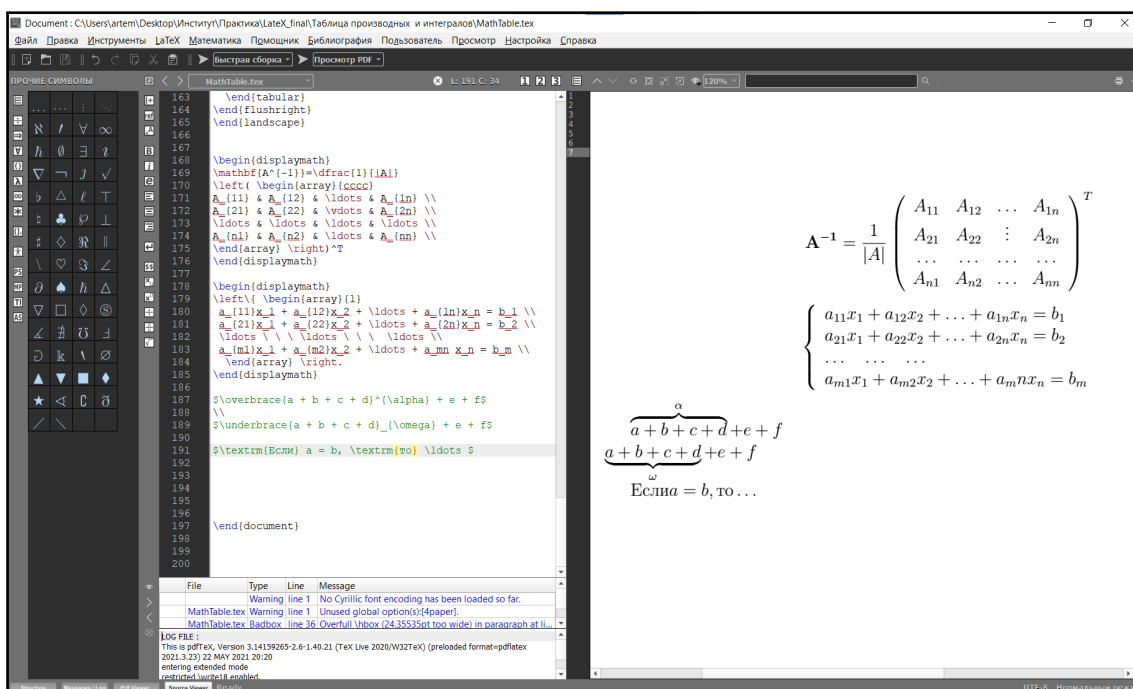


Рис. 4: "Крупные" математические объекты

Что же касается спец символов, в  $\text{\LaTeX}$ 'е их огромное количество, (к счастью) но раз уж речь идет о математике, то давайте попробуем собрать определение последовательности на языке " $\varepsilon \Delta$ "

$$\lim_{x \rightarrow x_0} f(x) = A \Leftrightarrow \forall \varepsilon > 0, \exists \delta > 0, \forall x \ 0 < |x - x_0| < \delta \Rightarrow |f(x) - A| < \varepsilon$$

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

$$\lim_{x \rightarrow \infty} \left(1 + \frac{1}{x}\right)^x = e$$

## 2.4 16.04.21-28.04.21

Для работы с графикой мы решили рассмотреть два пакета: `{graphicx}` и `{tikz}` первый служит для вставки растровых или векторных изображений в текст изображений в текст, а второй позволяет выполнять построение различных геометрических фигур, блок-схем, а также графиков некоторых функций, что представляет гораздо больший интерес. Начнем с пакета `{graphicx}`. Для нас нужно подключить его в преамбуле документа:

`\usepackage {graphicx }`

`{pictures/ }` Указываем название каталога где будут лежать изображения. (Он должен находиться в том же каталоге что и сам документ) Данная опция является необязательной, можно просто разместить все изображения в том же каталоге что и документ.

`\DeclareGraphicsExtensions{.pdf,.png,.jpg}` Указываем какие типы файлов будем использовать. Векторные изображения также поддерживаются.

Рассмотрим вставку изображений:

`\ begin {figure}[h!]` "Объявляем начало" изображения, в квадратных скобках указываем позицию изображения, "h!" обозначает, что изображение будет вставлено сразу после текста.

`\ setlength { \ fboxsep } {0pt }` размер полей вокруг изображения

`\ setlength { \ fboxrule } { 1pt }` ширина рамки

`\ fbox { \ includegraphics [ width=15cm,height=9cm ]{ Matrix 1 }`  
`}` задаем размеры изображения и указываем название файла (файл должен лежать в одной папке с документом)

`\ caption { "Крупные' математические объекты }|` Подпись под изображением

`\ end { figure } "Конец"`

Если подпись и рамка не требуются, то достаточно только строчки `\ fbox { \ includegraphics [ width=15cm,height=9cm ]{ Matrix 1 } }`. Вместо указания размеров в сантиметрах можно использовать команду `scale`.

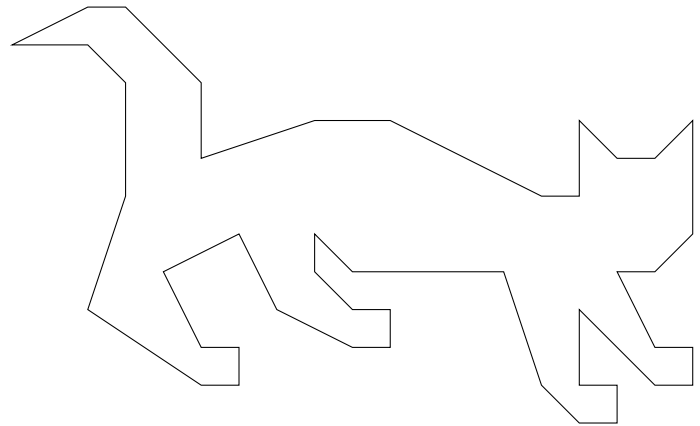
Итак, перейдем к пакету `{tikz}`. Его базовая функция - начертание фигур по их координатам.



```

\ begin{tikzpicture}
\draw(0,0)–(1,0)–(1.5,-0.5)–(1.5,-2)–
(1,-3.5)–(2.5,-4.5)–(3,-4.5)
–(3,-4)–(2.5,-4)–(2,-3)
–(3,-2.5)–(3.5,-3.5)–(4.5,-4)
–(5,-4)–(5,-3.5)–(4.5,-3.5)
–(4,-3)–(4,-2.5)–(4.5,-3)
–(6.5,-3)–(7,-4.5)–(7.5,-5)–
(8,-5)–(8,-4.5)–(7.5,-4.5)
–(7.5,-3.5)–(8.5,-4.5)–(9,-4.5)
–(9,-4)–(8.5,-4)–(8,-3)–(8.5,-3)
–(9,-2.5)–(9,-1)–(8.5,-1.5)
–(8,-1.5)–(7.5,-1)–(7.5,-2)
–(7,-2)–(5,-1)–(4,-1)–(2.5,-1.5)
–(2.5,-0.5)–(1.5,0.5)–(1,0.5)–cycle;
\ end { tikzpicture }

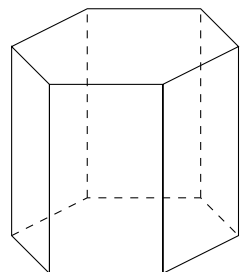
```



```

\ begin{tikzpicture}
\draw(0,0)–(1,0.5)–(2.5,0.5)–(3,0)–(2,-0.5)
–(0.5,-0.5)–(0,0)–(0,-2.5)–(0.5,-3)–(0.5,-0.5)–(0.5,-3)–(2,-3)
–(2,-0.5)–(2,-3)–(3,-2.5)–(3,0);
\ draw[dashed](0,-2.5)–(1,-2)–(2.5,-2)–(3,-2.5);
\ draw[dashed](1,-2)–(1,0.5);
\ draw[dashed](2.5,-2)–(2.5,0.5);
\ end { tikzpicture }

```

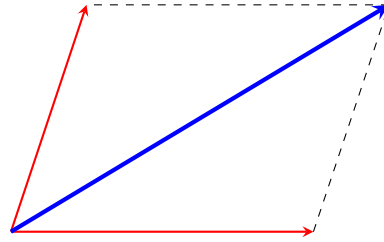


Все что нужно, это указывать координаты. При необходимости построить сложную фигуру можно строить несколько линий. Имеется возможность строить пунктирные линии, стрелки, также можно окрашивать в различные цвета, регулировать толщину.

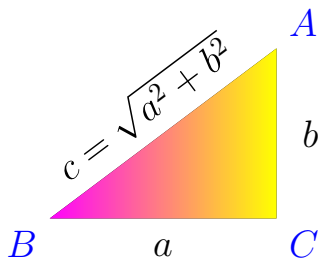
```

\begin{tikzpicture}[>=stealth]
\draw[thick, ->, red](0,0)–(4,0);
\draw[thick, ->, red](0,0)–(1,3);
\draw[ultra thick, ->, blue](0,0)–(5,3);
\draw[dashed](4,0)–(5,3)–(1,3);
\end{tikzpicture}

```



Добавление подписей к прямым и углам

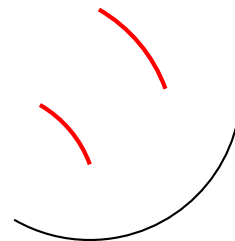


Окружности и дуги

```

\begin{tikzpicture}
\draw[Red,ultra thick](1,1) arc (20:60:2);
\draw[Red,ultra thick](0,0) arc (20:60:1.5);
\draw[thick](2,1) arc (0:-120:2);
\end{tikzpicture}

```

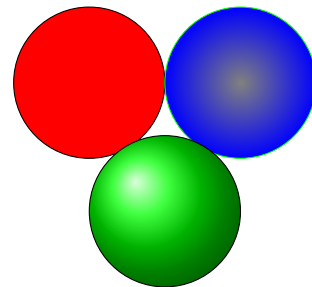


В квадратных скобках указываются дополнительные параметры (как и для любых линий), затем указываются координаты центра, а затем название фигуры (в нашем случае это дуга) далее следуют градусные меры начала и конца, длина радиуса.

```

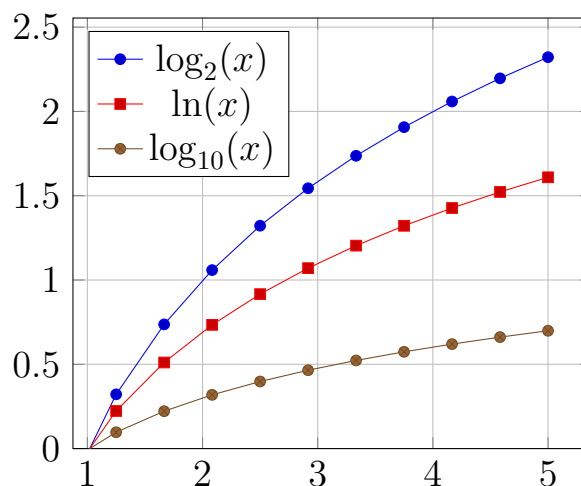
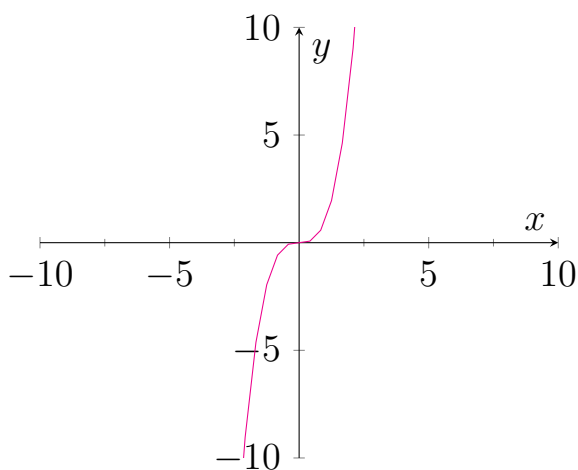
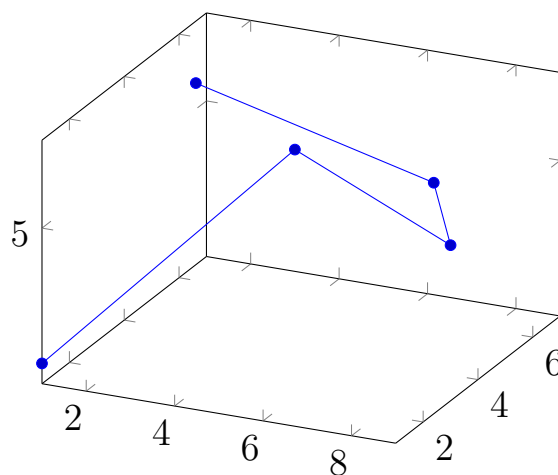
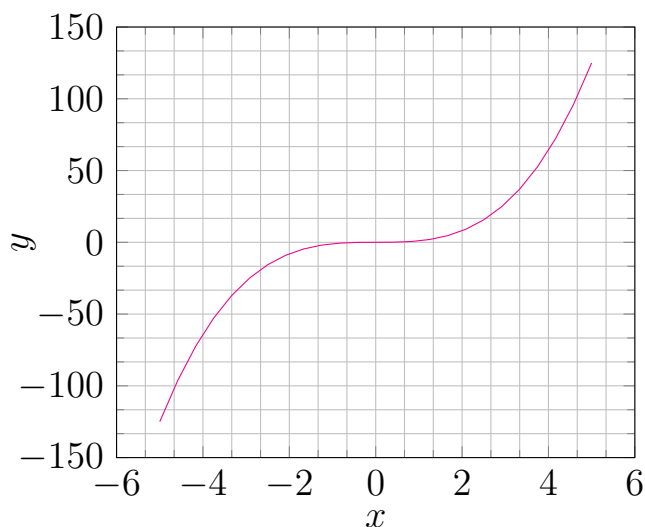
\begin{tikzpicture}
\draw[fill=Red] (0,0) circle (1);
\draw[fill=green, outer color=Blue] (2,0) circle (1);
\draw[fill=green] (1,-1.7) circle (1);
\end{tikzpicture}

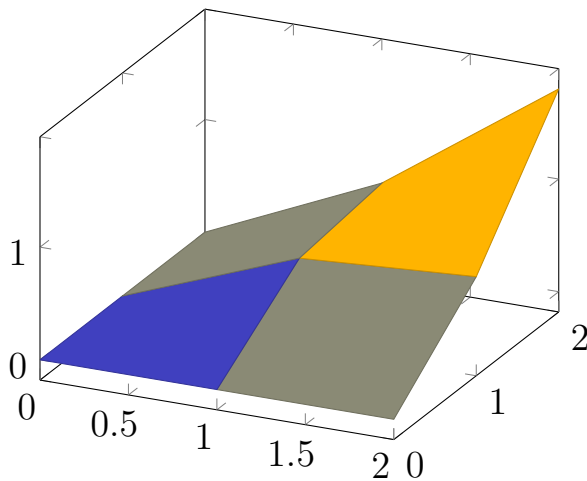
```



В пакете **{tikz}** имеется огромное количество готовых фигур, однако перечислять их мы не будем, ибо на это ушло бы слишком много времени. Однако мы не можем не показать использование данного пакета для построения графиков функций. (как предустановленных, так и при помощи таблицы значений)

Кубическая парабола





Как можно заметить, из последнего примера пакет `textbf TikZ` позволяет выполнять построение 3D объектов. Однако, мы не стали затрагивать его слишком подробно, ибо не обладаем столь глубокими познаниями в геометрии.

## 2.5 29.04.21-14.05.21

Приступили к составлению финального отчета. Выравнивание текста выполняется при помощи окружений `“left”` `“center”` `“right”`. Окружения это особый вид команд, позволяющий использовать специальные правила оформления больших частей текста или выполнять построение различных объектов. Окружения влияют на текст заключенный между командами начала и конца (`“\begin{имя окружения}”` `“\end{имя окружения}”` окружения и могут быть вложенными. Очень полезным для нас оказались окружения `“tabular”` и `“table”` (отличаются тем что в `“table”` отсутствует разметка и `“tabular”` может быть “вложен” в `“table”`) которое должно служить для составления таблиц. Однако пригождаются и для других целей:

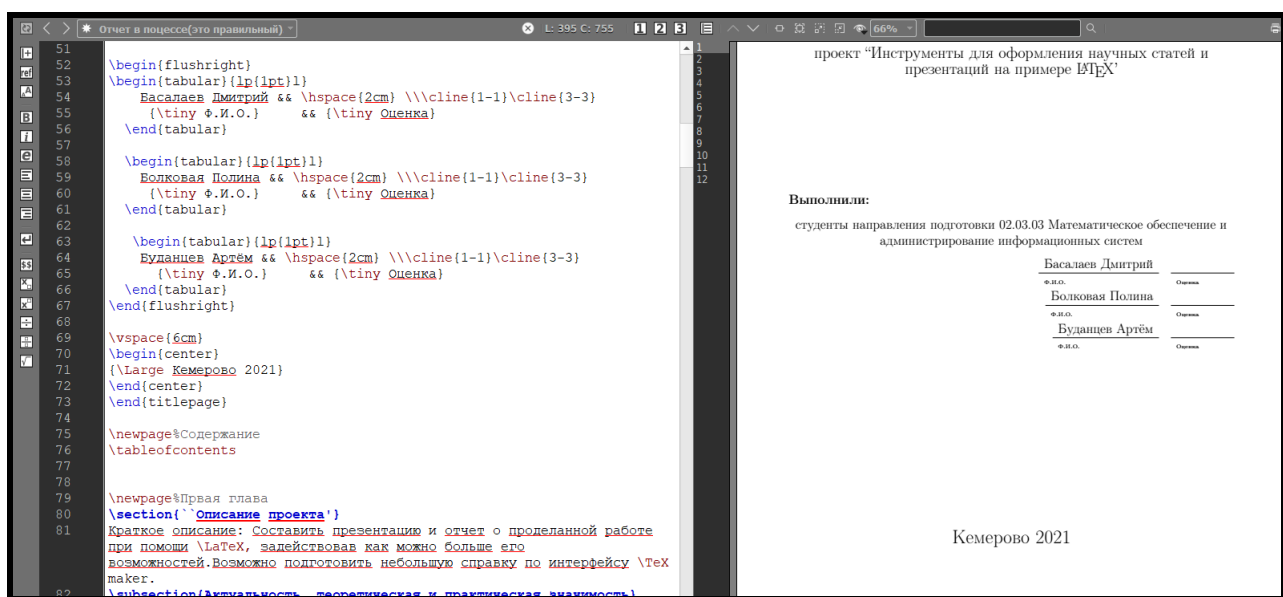


Рис. 5: Например на титульном листе

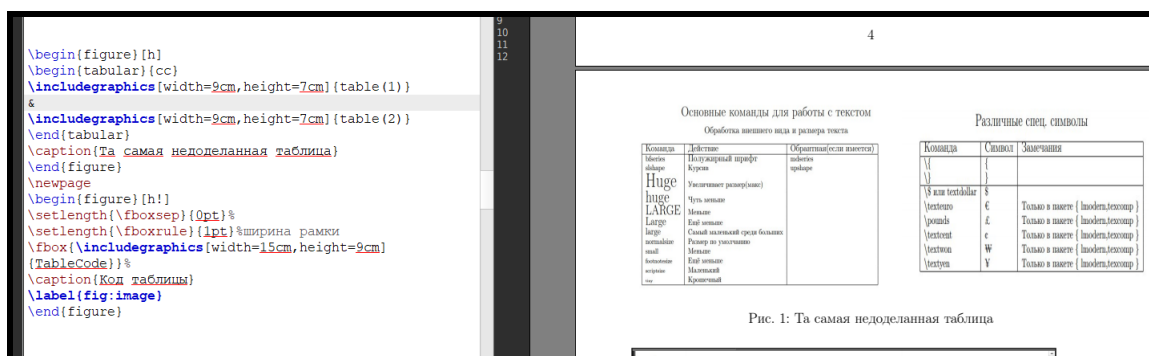


Рис. 6: Также полезно при вставке изображений



Рис. 7: С его помощью даже можно рисовать таблицы

Ещё одно “Простое окружение” *itemize* служит для составления списков. (+ по сравнению с Word: можно вручную вписывать маркеры)

```
\begin{itemize}
  \item[а сюда можно вписывать маркеры
    (по умолчанию это точки)]
    Level 0 Item 0
  \item Level 0 Item 1
  \begin{itemize}
    \item Level 1 Item
    \begin{itemize}
      \item Level 2 Item
      \begin{itemize}
        \item Level 3 Item
      \end{itemize}
    \end{itemize}
  \end{itemize}
\end{itemize}
```

Также есть нумерованный список:

```
\begin{enumerate}
  \item первый элемент первого уровня содержит список
    \begin{enumerate}
      \item элемент списка второго уровня
      \item второй элемент списка второго уровня
    \end{enumerate}
  \item второй элемент первого уровня
\end{enumerate}
```

Также мы ознакомились с окружением “*scope*” которое является вспомогательным к “*tikzpicture*” и служит для работы с tikz рисунком как с единым объектом. (масштабирование, позиционирование)

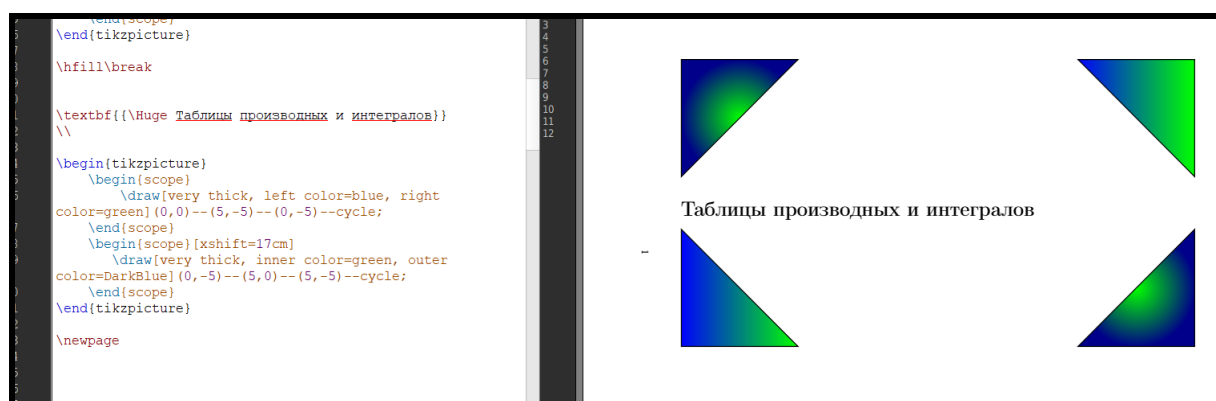


Рис. 8: Однако в основном отчете оно нам не пригодились

Эта строка внутри “*minipage*”

В полном соответствии со своим

названием окружение ***minipage*** создает страницу внутри страницы, которая может содержать свои собственные сноски, абзацы, а также окружения ***array***, ***tabular*** и ***multicols***. Кроме того, ***minipage*** можно использовать внутри плавающих объектов, создаваемых окружениями ***figure*** и ***table***, добиваясь весьма интересных эффектов.

Параметр pos может принимать одно из трех значений — c, t или b и управляет вертикальным выравниванием блока по отношению к базовой линии окружающего текста — по центру, по верхнему и по нижнему краю соответственно. height определяет высоту, а width — ширину блока. Параметр contentpos управляет расположением содержимого по вертикали внутри блока, и принимает уже знакомые значения c, t и b.

Также крайне полезным оказалось окружение ***textbfverbatim*** в зоне его действия текст не воспринимается как команды, все пробелы и отступы отображаются “как есть” а также не учитывается разметка страницы. При помощи этого окружения очень легко вставить в документ код практически на любом языке программирования. (имеются даже опции для подсветки синтаксиса)

<sup>1</sup>

Также в **L<sup>A</sup>T<sub>E</sub>X** можно реализовать одно из важных преимуществ pdf документов - ссылки. Если подключить пакет ***hyperref*** то автособираемое автоматически превратится в набор ссылок (что очень удобно) Также можно рассавлять ссылки вручную:

<sup>1</sup>Кстати тут есть сноски, не знаю стоит ли это упоминать (ибо мы ими опять таки не пользовались)

смотреть здесь!

```
\hypertarget{d6(а это уникальный номер ссылки)}  
{Это якорь, сюда будет указывать ссылка}
```

```
\hyperlink{d6(это уникальный номер ссылки)}{А это текст самой ссылки}
```

Мы предстпуили к созданию презентации, в  $\text{\LaTeX}$  за это класс документа **{beamer}**, за разделение на слайды отвечает окружение **frame**. Код слайда обычно выглядит так:

```
\usebackgroundtemplate{\includegraphics  
[width=\paperwidth,height=\paperheight]  
{Если хотим поставить изображение на фон}}  
\begin{frame}{заголовок слайда}  
текст  
\end{frame}
```

Что касается текста, то все что можно сделать в **article**<sup>2</sup> можно сделать и в **beamer**<sup>3</sup>. Однако есть несколько уникальных окружений (только для презентаций)

```
\begin{block}{Заголовок блока}  
Текст  
\end{block}
```

Это окружение для выделения важной информации. Выглядит так:



Рис. 9: Есть несколько видов блоков

Также в **beamer** есть анимации переходов и появления объектов на слайде.

<sup>2</sup>Класс для составления обычных документов

<sup>3</sup>Класс для составления презентаций

## Анимации переходов

- `blindshorizontal` — слайд «разрезается» вертикальными полосами;
- `blindsvertical` — слайд «разрезается» горизонтальными полосами;
- `boxin` — старый слайд «стягивается» в точку по центру экрана;
- `boxout` — новый слайд «растягивается» из точки по центру;
- `dissolve` — слайд сменяется мозаикой;
- `glitter` — смесь `dissolve` с `wipe`;
- `splitverticalin` «стягивается» сверху и снизу;
- `splitverticalout` «растягивается» сверху и снизу;
- `wipe` — следующий слайд «выезжает» слева.

Рис. 10: Список анимаций

## 2.6 15.05.21-

Приступили к составлению этого отчета и презентации.

## 3 Выводы

Поначалу работать в не WYSIWYG редакторе было трудно и непривычно. Простые действия, которые в том же Ms Word делаются “двумя щелчками мыши”, здесь могли требовать подключения отдельных пакетов и ввода нескольких команд. Но в тоже время, в  $\text{LaTeX}$  мы имели более “жесткую” и оттого более удобную и практичную разметку страниц, простые в составлении таблицы (в которых точно ничего “не поедет”), быстрое и удобное составление оглавления. Легко настраиваемые списки с возможностью расставления маркеров вручную, тонко настраиваемую вставку изображений. Ну и конечно же быстрый ввод математических формул и объектов, построение качественных графиков и диаграм. Учитывая все вышеперечисленное мы настоятельно рекомендуем присмотреться к  $\text{LaTeX}$  в качестве основного редактора для документов и презентаций.

## 4 Литература

1. Львовский С. М. Набор и вёрстка в системе  $\text{LaTeX}$ . — М.: МЦНМО, 2006. — 448 с.
2. Котельников И. А., Чеботаев П. З.  $\text{LaTeX}$  по-русски. — СПб.: «Корона-Век», 2011. — 496 с.



3. А.А. Жидков Интерактивные презентации в системе L<sup>A</sup>T<sub>E</sub>X Учебно-методическое пособие
4. Курс «Документы и презентации в LaTeX (Introduction to L<sup>A</sup>T<sub>E</sub>X)» на сайте [coursera.org-https://www.coursera.org/learn/latex/home/welcome](https://www.coursera.org/learn/latex/home/welcome)
5. Материалы с сайта <https://ru.overleaf.com>