

# $\text{T}_\text{E}_\text{X}$ , $\text{L}_\text{A}_\text{T}_\text{E}_\text{X}$ , $\text{A}_\text{M}_\text{S}-\text{T}_\text{E}_\text{X}$ , $\text{A}_\text{M}_\text{S}-\text{L}_\text{A}_\text{T}_\text{E}_\text{X}$ , $\text{M}_\text{i}_\text{k}\text{T}_\text{E}_\text{X}$ , $\text{P}_\text{y}\text{t}\text{h}\text{o}\text{n}\text{T}_\text{E}_\text{X}$ – как основа автоматической системы генерации банка задач

Артем Ю. Золотаревский<sup>1</sup>, Павел Е. Рябов<sup>1</sup>

<sup>1</sup>Финансовый университет при Правительстве Российской Федерации



artyom@zolotarevskiy.ru, PERyabov@fa.ru

Международная научно-практическая конференция  
«Современная математика и концепции  
инновационного математического образования»  
Москва, Финансовый университет

## План

- Оригинальные книги
- MiKTeX
- PythonTeX – как основа автоматической системы генерации банка задач


# План доклада

## План

- Оригинальные книги
- **МiKTeX**
- PythonTeX – как основа автоматической системы генерации банка задач

## План

- Оригинальные книги
- MiKTeX
- PythonTeX – как основа автоматической системы генерации банка задач

 *Дональд Э. Кнут* The TeX-book / Пер. с англ. М. В. Лисиной. – Протвино: АО RDTeX, 1993. – 592 с. – ISBN 5-900614-01-8.

## Все про TeX

# Все про TeX

ДОНАЛЬД Е. КНУТ *Станфордский Университет*



Иллюстрации  
ДВОРНИК БИББИ

Перевод с английского и  
подготовка оригинал-макета  
М. В. ЛИСИНОЙ  
под редакцией  
С. В. КЛИМЕНКО  
С. Н. СОКОЛОВА

Издательство АО RDTeX  
Протвино, 1993 г.

## Описание

Самое полное руководство по использованию системы TeX, написанное ее создателем — Дональдом Э. Кнутом. TeX — это новый этап в компьютерном наборе.



*Дональд Эрвин*

*Кнут*

Система TeX особенно полезна, когда подготавливаемые к печати документ, статья или книга содержат много математических формул, и позволяет создать печатную продукцию высокого качества. В данной книге изложены базовые знания по системе TeX в ее оригинальном виде, придуманной автором этой книги Дональдом Кнутом.

 *Дональд Э. Кнут* Все про METAFONT. – М.: «Вильямс», 2003. – 384 с. – ISBN 5-8459-0442-0.

## Все про METAFONT

### Все про METAFONT

ДОНАЛЬД Э. КНУТ

*Станфордский университет*



Москва · Санкт-Петербург · Киев  
2003

## Описание



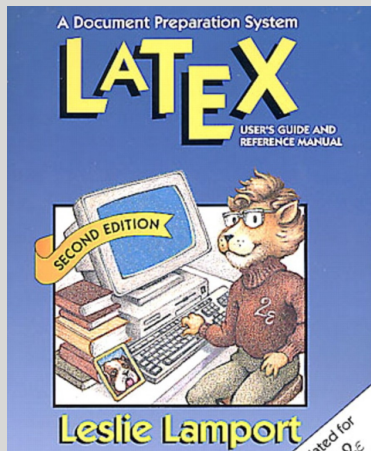
*Дональд Эрвин*

*Кнут*

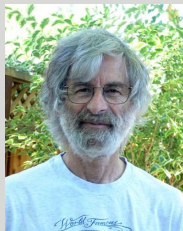
METAFONT – язык программирования, используемый для разработки векторных шрифтов. Также это название интерпретатора, выполняющего код METAFONT. Этот язык программирования был разработан Дональдом Кнутом как дополнение к его типографской системе T<sub>E</sub>X. Одной из особенностей METAFONT является то, что все очертания символов определяются с помощью развитых геометрических описаний, например, возможно определить какую-либо точку как пересечение отрезка и кривой Безье.

-  *Leslie Lamport*  $\text{\LaTeX}$ : a document preparation system. – New York: Addison-Wesley Publishing Company, Inc., 1994. – 273 с. – ISBN 0-201-52983-1.

$\text{\LaTeX}$



## Описание



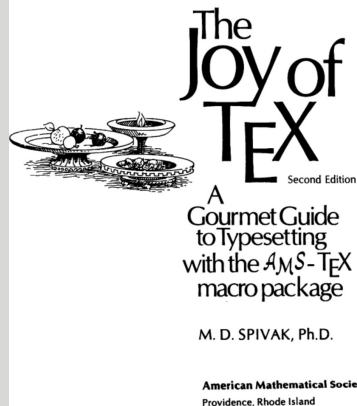
*Leslie Lamport*  
как  $\text{\LaTeX}$ .

$\text{\LaTeX}$  – наиболее популярный набор макрорасширений (или макропакет) системы компьютерной вёрстки  $\text{\TeX}$ , который облегчает набор сложных документов. В типографском наборе системы  $\text{\TeX}$  форматируется традиционно

как  $\text{\LaTeX}$ . Важно заметить, что ни один из макропакетов для  $\text{\TeX}$ 'а не может расширить возможностей  $\text{\TeX}$  (всё, что можно сделать в  $\text{\LaTeX}$ 'е, можно сделать и в  $\text{\TeX}$ 'е без расширений), но, благодаря различным упрощениям, использование макропакетов

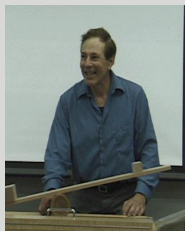
-  *Spivak, Michael* The Joy of T<sub>E</sub>X: A Gourmet Guide to Typesetting with the A<sub>M</sub>S-T<sub>E</sub>X Macro package, (1990).

## A<sub>M</sub>S-T<sub>E</sub>X, A<sub>M</sub>S-L<sub>A</sub>T<sub>E</sub>X



## Описание




A<sub>M</sub>S-L<sub>A</sub>T<sub>E</sub>X— набор макроспирений для L<sub>A</sub>T<sub>E</sub>X, разработанный Американским математическим обществом. A<sub>M</sub>S-L<sub>A</sub>T<sub>E</sub>X вытеснил набор расширений A<sub>M</sub>S-T<sub>E</sub>X, который был написан Майклом Спиваком.

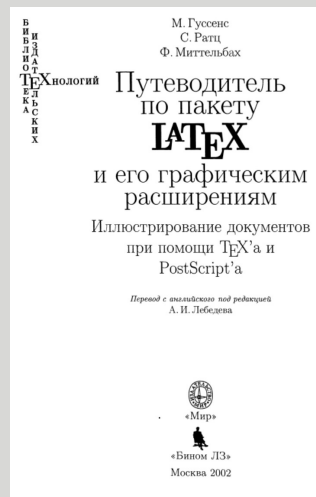






Майкл Дэвид

Спивак

Майкл Дэвид Спивак – американский математик, специализирующийся на дифференциальной геометрии. Книга «Теория Морса», написанная Джоном Милнором, была основана на конспектах лекций Спивака и Роберта Уэллса.

-  *Гуссенс М., Миттельбах Ф., Самарин А.* Путеводитель по пакету  $\text{\LaTeX}$  и его расширению  $\text{\LaTeX}2_{\epsilon}$ . — М.: «Мир», 1999. — 606 с. — ISBN 5-03-003325-4.
-  *Гуссенс М., Ратц С.* Путеводитель по пакету  $\text{\LaTeX}$  и его Web-приложениям. — М.: «Мир», 2001. — 604 с. — ISBN 5-03-003387-4.
-  *Гуссенс М., Ратц С., Миттельбах Ф.* Путеводитель по пакету  $\text{\LaTeX}$  и его графическим расширениям. — М.: «Мир», «Бином ЛЗ», 2002. — 621 с. — ISBN 5-03-003388-2.



-  *Балдин, Е. М.* Компьютерная типография LATEX / Е. М. Балдин. – СПб.: БХВ- Петербург, 2008. – 304 с.
-  *Львовский, С. М.* Набор и вёрстка в системе LATEX / С. М. Львовский. – М.: МЦНМО, 2003. – 448 с.
-  *Котельников, И. А.* LATEX по-русски / И. А. Котельников, П. З. Чеботаев. – Новосибирск: Сибирский Хронограф, 2004. – 496 с.
-  *Роженко, А. И.* Искусство верстки в LATEX'е / А. И. Роженко; Под ред. А. С. Алексеева. – Новосибирск: Изд. ИВМиМГ СО РАН, 2005. – 398 с.

Примерно через десять лет после начала работы над проектом системы METAFONT (создание шрифтов) и TEX (лучшая программа разбиения абзацев на строки) были стабилизированы (версия 2.7 для METAFONT и 3.1 для TEX). Кнут отошёл от активной разработки. В дальнейшем METAFONT и TEX модифицировались только с целью исправления ошибок. На текущий момент номер версии METAFONT равен 2.71828, а TEX — 3.141592. Кнут завещал, что после его смерти номера версий будут заморожены и равны числу  $e$  и числу  $\pi$  соответственно, а все оставшиеся неисправленные ошибки будут считаться особенностями реализации.

## МіК<sub>Т</sub>Е<sub>Х</sub>

МіК<sub>Т</sub>Е<sub>Х</sub>— открытый дистрибутив Т<sub>Е</sub>Х для платформы Windows. Одним из существенных достоинств МіК<sub>Т</sub>Е<sub>Х</sub> является возможность автоматического обновления установленных компонентов и пакетов.

Особенности последних версии МіК<sub>Т</sub>Е<sub>Х</sub>(2.7-2.9) является интегрированная поддержка Xe<sub>Т</sub>Е<sub>Х</sub>, Lua<sub>Т</sub>Е<sub>Х</sub>, MetaPost, pdf<sub>Т</sub>Е<sub>Х</sub>

В настоящее время в состав МіК<sub>Т</sub>Е<sub>Х</sub> включены:

- текстовый редактор Т<sub>Е</sub>Х-works;
- компиляторы Т<sub>Е</sub>Х, pdf<sub>Т</sub>Е<sub>Х</sub>, Xe<sub>Т</sub>Е<sub>Х</sub> и Lua<sub>Т</sub>Е<sub>Х</sub>;
- конверторы Т<sub>Е</sub>Х в PDF: Dvipdfm/Dvipdfmx; MetaPost
- полный набор общеиспользуемых макропакетов: L<sup>A</sup>T<sub>Е</sub>X, A<sub>M</sub>S-L<sup>A</sup>T<sub>Е</sub>X, Python<sub>Т</sub>Е<sub>Х</sub>
- средство просмотра Y<sub>a</sub>p

## WinEdt

WinEdt — это shareware-редактор для файлов ASCII и оболочка для Microsoft Windows. Основная сфера использования программы — это создание  $\text{T}_\text{E}\text{X}$ - и  $\text{L}\text{A}\text{T}_\text{E}\text{X}$ -документов; также широко используется для редактирования документов HTML и других текстовых файлов.

- MDI-интерфейс, поддерживающий управление включаемыми файлами в файле проекта WinEdt;
- Подсветка синтаксиса;
- Структура документа и оглавление отображаются в собираемом меню;
- Управление библиографией и метками;
- Настраиваемый запуск программ  $\text{L}\text{A}\text{T}_\text{E}\text{X}$ ,  $\text{T}_\text{E}\text{X}$ ,  $\mathcal{A}\mathcal{M}\mathcal{S}\text{T}_\text{E}\text{X}$ ,  $\text{PDFL}\text{A}\text{T}_\text{E}\text{X}$
- Доступ к математическим символам и средствам форматирования текста через кнопки графического интерфейса;

## PythonTeX – как основа автоматической системы генерации банка задач

- **Хранение данных**
- Панель управления
- Результаты генерации
- Жизненный цикл билета
- Возможные ошибки и способы их решения
- Примеры
- Предстоящие улучшения

## PythonTeX – как основа автоматической системы генерации банка задач

- Хранение данных
- **Панель управления**
- Результаты генерации
- Жизненный цикл билета
- Возможные ошибки и способы их решения
- Примеры
- Предстоящие улучшения

## PythonTeX – как основа автоматической системы генерации банка задач

- Хранение данных
- Панель управления
- **Результаты генерации**
- Жизненный цикл билета
- Возможные ошибки и способы их решения
- Примеры
- Предстоящие улучшения

## PythonTeX – как основа автоматической системы генерации банка задач

- Хранение данных
- Панель управления
- Результаты генерации
- **Жизненный цикл билета**
- Возможные ошибки и способы их решения
- Примеры
- Предстоящие улучшения

## PythonTeX – как основа автоматической системы генерации банка задач

- Хранение данных
- Панель управления
- Результаты генерации
- Жизненный цикл билета
- **Возможные ошибки и способы их решения**
- Примеры
- Предстоящие улучшения

## PythonTeX – как основа автоматической системы генерации банка задач

- Хранение данных
- Панель управления
- Результаты генерации
- Жизненный цикл билета
- Возможные ошибки и способы их решения
- **Примеры**
- Предстоящие улучшения

## PythonTeX – как основа автоматической системы генерации банка задач

- Хранение данных
- Панель управления
- Результаты генерации
- Жизненный цикл билета
- Возможные ошибки и способы их решения
- Примеры
- **Предстоящие улучшения**

# PythonTeX – как основа автоматической системы генерации банка задач

Описано устройство работы разработанной системы автоматической генерации банка задач для контрольных работ, экзаменационных билетов различных разделов математических дисциплин, основанной на открытых приложениях «MikTeX», «PythonTeX» и «Jupyter Notebook».

Главной структурной единицей является  $\text{T}_\text{E}\text{X}$  файл, в котором хранится Python-код для параметризации, формулировка задачи и ее решение. Эти данные должны располагаться в так называемых блоках «окружения» (environment). Три блока окружения «pycode», «problem», «solution» определяют задачу.

Данный  $\text{T}_\text{E}\text{X}$  файл может содержать неограниченное кол-во задач.


```
\begin{pycode}
# здесь Python-параметризация
\end{pycode}

\begin{problem}
Здесь формулировка задачи
\end{problem}

\begin{solution}
А здесь решение
\end{solution}
```

## Пример параметризации простой задачи

```


\begin{pycode}
from random import randint
import math

big_radius = randint(50, 100)
point_count = randint(2, 10)
small_radius = randint(10, big_radius - 10)

p = round(1 - ((math.pi * big_radius ** 2) - (math.pi * small_radius ** 2)) / (math.pi * big_radius ** 2)) ** point_count, 4)
\end{pycode}

\begin{problem}
В круг радиуса  $\text{py}\{big\_radius\}$  наудачу бросаются  $\text{py}\{point\_count\}$  точки.\\
Найдите вероятность того, что расстояние от центра круга до ближайшей точки будет не больше  $\text{py}\{small\_radius\}$ 
\end{problem}

\begin{solution}
Вероятность  $\mathbb{P} = \text{py}\{p\}$ 
\end{solution}

```

## Результат простой параметризации

В круг радиуса 62 наудачу бросаются 4 точки.

Найдите вероятность того, что расстояние от центра круга до ближайшей точки будет не больше 10

**Решение.** Вероятность  $\mathbb{P} = 0.1001$

Двое договорились о встрече между 11 и 14 часами утра, причем договорились ждать друг друга не более 40 минут. Считая, что момент прихода на встречу выбирается каждым наудачу в пределах указанного часа, найдите вероятность того, что встреча состоится.

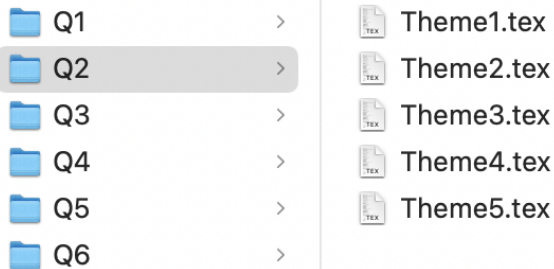
**Решение.** Вероятность  $\mathbb{P} = 0.3951$

На отрезке  $[2, 96]$  случайным образом выбираются два действительных числа:  $X$  и  $Y$ . Найдите вероятность того, что расстояние между  $X$  и  $Y$  будет не меньше 72.

**Решение.** Вероятность  $\mathbb{P} = 0.0548$

# Хранение данных - Разделение задач по темам

Задачи для генерации одного экзаменационного вопроса предлагается разделять по темам на разные  $\text{T}_\text{E}_\text{X}$  файлы, которые должны располагаться в соответствующих папках вопросов  $Q_1, Q_2, \dots, Q_N$ .



## Шаблон билета

Итоговые билеты строятся на основе единого  $\text{T}_\text{E}\text{X}$ -файла шаблона, в котором определены все необходимые зависимости, а также есть шапка и футер.

Панель управления генератора реализована в виде документа Jupyter Notebook.

Для генерации банка задач нужно лишь выполнить функцию «generate\_exam» с желаемыми параметрами.

```
1 from taskgen import *
2 from taskgen.generator import *
3 from taskgen.html2pdf import *

1 # начало нумерации билетов
2 start_numeration = 100
3 # кол-во генерируемых вариантов
4 variant_count = 1
5 # детерминированная генерация билетов
6 deterministic = True

1 %%time
2 generate_exam(start_numeration, variant_count, deterministic)
```

# Панель управления - другие полезные функции





```
gen_variant(variant_number = 1, deterministic = True, task_number_for_deterministic = 0)

compile_file(filename = 'Theme1', folder = './QUESTIONS/Q1/')

html2pdf(os.path.join(os.getcwd(), 'RESULTS', 'html', 'only_problems'), \
         os.path.join(os.getcwd(), 'RESULTS', 'pdf', 'only_problems'), in_one_page = True)
```

# Результаты генерации

Главным результатом программы являются 2 pdf файла: в одном хранятся объединенные готовые к печати варианты билетов, а в другом ответы к ним.

-  only\_problems
-  problems\_with\_answers
-  only\_problems\_merged.pdf
-  problems\_with\_answers\_merged.pdf

## Пример сгенерированного варианта

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ БЮДЖЕТНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ  
РОССИЙСКОЙ ФЕДЕРАЦИИ»  
(ФИНАНСОВЫЙ УНИВЕРСИТЕТ)

Департамент анализа данных и машинного обучения

*Дисциплина: «Теория вероятностей и математическая статистика»*

*Направление подготовки: 01.03.02 «Прикладная математика и информатика»*

*Профиль: «Анализ данных и принятие решений в экономике и финансах»*

*Факультет информационных технологий и анализа больших данных*

*Форма обучения очная*

*Учебный 2020/2021 год, 3 семестр*

### Билет № 100

**1. (10)** Дайте определение случайной величины, которая имеет  $\chi^2$ -распределение с  $n$  степенями свободы. Запишите плотность  $\chi^2$ -распределения. Выведите формулы для математического ожидания  $E(X)$  и дисперсии  $\text{Var}(X)$  хи-квадрат распределения с  $n$  степенями свободы.

Найдите:

- а)  $P(\chi_{24}^2 > 13,84)$ , где  $\chi_{24}^2$  – случайная величина, которая имеет  $\chi^2$ -распределение с 24 степенями свободы;  
б) 90 (верхнюю) точку  $\chi_{0,9}^2(7)$  хи-квадрат распределения с 7 степенями свободы.

**2. (10)**  $Z_{\text{набл.}} = -0.34273$ ;  $K_\alpha = (-\infty; -2.32635) \cup (2.32635; +\infty)$ ;  $PV = 0.7318$ ;  $\beta(\mu_1) = 0.83179$ ;  $W(\mu_1) = 0.16821$ .

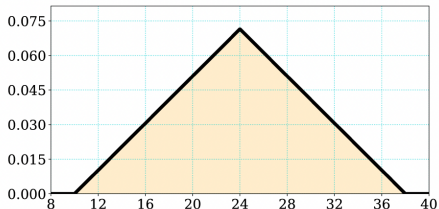
Пусть  $\vec{x}$  – реализация случайной выборки  $X_1, X_2, \dots, X_{1200}$  из нормального распределения  $N(\mu; 3.300000000000003^2)$  (файл ds.100-2.csv).

Проверяется на уровне значимости  $\alpha = 0.02$  основная гипотеза  $H_0: \mu = 1.01$  против альтернативной гипотезы  $H_1: \mu \neq 1.01$ .

- 1) Найдите значение статистики критерия  $z = Z_{\text{набл.}} = Z(\vec{x})$ , критическое множество  $K_\alpha$  и на 2.0 уровне значимости проверьте  $H_0$ .  
2) Найдите  $P$ -значение критерия и сделайте выводы.  
3) Найдите вероятность ошибки второго рода  $\beta$  и мощность критерия  $W$  для  $\mu_1 = 0.88$ .

**3. (10)** Случайные величины  $X$  и  $Y$  независимы и имеют равномерное распределение на отрезке  $[5; 19]$ . Для случайной величины  $Z = X + Y$  найдите: 1) функцию распределения  $F_Z(x)$ ; 2) плотность распределения  $f_Z(x)$  и постройте график плотности; 3) значение  $F_Z(31,14)$ .

## Пример сгенерированного варианта



4. (10) В круг радиуса 57 наудачу бросаются 3 точки.  
Найдите вероятность того, что расстояние от центра круга до ближайшей точки будет не больше 45

5. (10) Случайный вектор  $(X, Y)$  имеет плотность распределения  $f_{X,Y}(x, y) = \frac{3e^{-2x^2 - 8xy + 6y - \frac{25y^2}{2} + 2y - \frac{181}{18}}}{\pi}$ .

Найдите:

- 1) математическое ожидание  $E(X)$ ;
- 2) математическое ожидание  $E(Y)$ ;
- 3) дисперсию  $D(X)$ ;
- 4) дисперсию  $D(Y)$ ;
- 5) ковариацию  $Cov(X, Y)$ ;
- 6) коэффициент корреляции  $\rho(X, Y)$ .

6. (10) По содержимому в файле ds.100-6.csv реализации случайной выборки из двумерного нормального распределения

$N\left(\begin{pmatrix} E(X) \\ E(Y) \end{pmatrix}; \begin{pmatrix} \text{Var}(X) & \text{Cov}(X, Y) \\ \text{Cov}(X, Y) & \text{Var}(Y) \end{pmatrix}\right) = N\left(\begin{pmatrix} -3 \\ 3 \end{pmatrix}; \begin{pmatrix} \sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \sigma^2 \end{pmatrix}\right)$  с неизвестными параметрами  $\rho \in (-1; 1)$  и  $\sigma > 0$ :

- 1) запишите логарифм функции правдоподобия,  $\ln L(\rho, \sigma)$ ;
- 2) найдите оценки максимального правдоподобия  $\hat{\rho}$  и  $\hat{\sigma}$ .

## Промежуточные файлы

Система также создает:

- исходные PythonTeX файлы вариантов, собранные на основе типовых задач;
- скомпилированные TeX файлы вариантов, т.е. содержащие только результаты параметризации без Python-кода;
- html версии вариантов, используемые для «умного» масштабирования;
- pdf версии вариантов, готовые к печати;

## Шаг 1. Генерация PythonTeX файла билета

Из файлов с типовыми задачами для каждого вопроса нового варианта выбирается случайно или детерминировано (в зависимости от параметров запуска) задача и создается PythonTeX-файл билета, содержащий исполняемый код параметризации, условие и решение.

## Шаг 2. Компиляция PythonTeX файла

Сгенерированный на прошлом этапе исходный файл компилируется и на его основе создаются 2 TeX файла с результатами выполнения Python-кода: один с решением, а другой без.

## Шаг 3. Конвертация tex2html

Созданные TeX файлы конвертируются в html. Для этого используется входящий в дистрибутив «MikTeX» инструмент командной строки «htlatex». Все pdf изображения конвертируются в png с разрешением 300 DPI. Для корректного отображения TeX в Web'е используется кросс-браузерная JavaScript библиотека MathJax. А благодаря специальному конфигурационному файлу шаблона страницы можно доопределить необходимые стили html-документа или параметры данной библиотеки.

## Шаг 3. Конвертация tex2html. Конфигурационный файл html-страницы

```
1 \Preamble{xhtml,mathml}
2 \Configure{VERSION}{}
3 \Configure{DOCTYPE}{\HCode{<!DOCTYPE html>\Hnewline}}
4 \Configure{HTML}{\HCode{<html>\Hnewline}}{\HCode{\Hnewline</html>}}
5 \Configure{@HEAD}{}
6 \Configure{@HEAD}{\HCode{<meta charset="UTF-8" /\Hnewline}}
7 \Configure{@HEAD}{\HCode{<meta name="generator" content="TeX4ht" /\Hnewline}}
8 \Configure{@HEAD}{\HCode{<link
9     rel="stylesheet" type="text/css"
10    href="\expandafter\csname aa:CssFile\endcsname" /\Hnewline}}
11 \Configure{@HEAD}{\HCode{<script type="text/javascript"\Hnewline
12 src="http://cdn.mathjax.org/mathjax/latest/MathJax.js?config=TeX-AMS-MML_HTMLorMML"\Hnewline
13 ></script>\Hnewline}}
14 \Configure{@HEAD}{\HCode{<style type="text/css">\Hnewline
15 .MathJax_MathML {text-indent: 0;}\Hnewline
16 </style>\Hnewline}}
17 \begin{document}
18 \EndPreamble
```

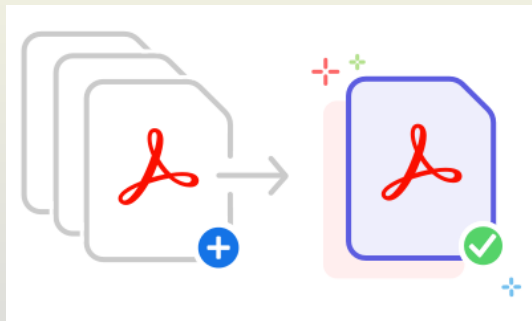
## Шаг 4. Конвертация html2pdf

Созданные html файлы конвертируются в pdf. При этом вариант занимает всю страницу, т.е. происходит автоматическое масштабирование.

Для реализации данного функционала используется headless-версия браузера Google Chrome со встроенной в нее системой печати веб-страниц в pdf файл. С помощью библиотеки автоматизации браузеров «Selenium» было организовано взаимодействие с внутренним API Хрома. Дождавшись загрузки html страницы и завершения рендеринга формул, с помощью разработанного алгоритма производится расчет коэффициента масштабирования и в конечном счете создается необходимый pdf файл варианта.

## Шаг 5. Объединение pdf файлов

Созданные pdf файлы объединяются в итоговый единый документ, готовый к печати. В результате имеем отдельный файл со всеми вариантами, а также отдельный файл со всеми решениями.



# Возможные ошибки и способы их решения

Самыми популярными будут ошибки синтаксические при наборе  $\TeX$ -верстки или Python-параметризации.

Для компиляции и отладки файлов с типовыми задачами предлагается использовать функцию «`compile_file`» в панели управления генератора. В случае неудачи, она выводит последние 50-й строк лог-файла, в которых, как правило, и содержится сообщение об ошибке. Кроме этого, она удаляет временные файлы, оставшиеся с предыдущей компиляции PythonTeX. Такой подход позволяет избежать известную всем проблему кеширования, когда изменения в исходный файл вроде бы внесены, а результат компиляции не меняется.

# Что мы получили?

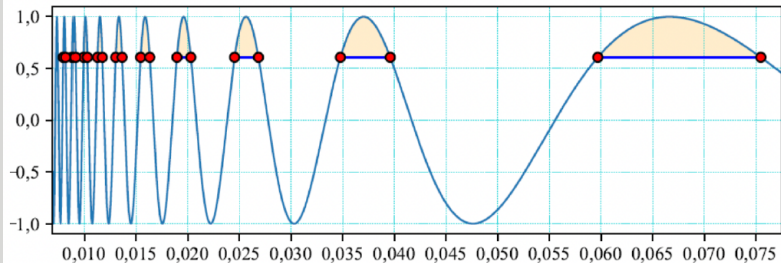
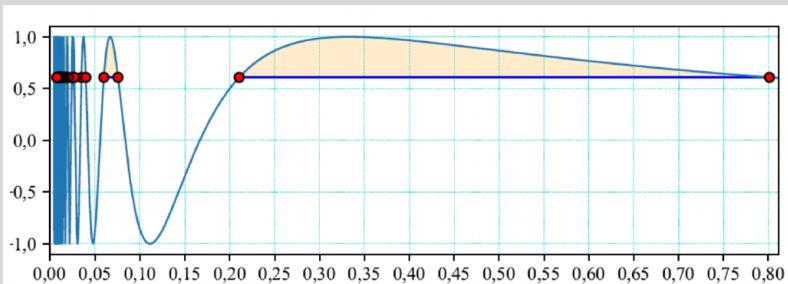
Разработанное автором решение, грамотно объединяющая издательскую систему верстки «MikTeX» и вычислительную среду «PythonTeX», позволяет преподавателям сфокусироваться на самом процессе параметризации банка задач, а не способе его подготовки. При этом все компоненты являются открытыми и признанными сообществом. В результате мы имеем бесплатный мощный инструмент, отлично решающий поставленную задачу.

# Пример использования matplotlib

Графическая иллюстрация

```
1 from matplotlib.pyplot import plot
2 from matplotlib.ticker import MultipleLocator
3 import locale
4
5 f = lambda x: np.sin(np.pi / (6 * x))
6 x1k = k / (np.pi - np.arcsin(a) + 2 * np.pi * np.array(range(0, 11)))
7 x2k = k / (np.arcsin(a) + 2 * np.pi * np.array(range(0, 11)))
8 x = np.linspace(0.005, 1.1, 100000)
9
10 rc('axes.formatter', use_locale=True), rc('font', family='Times New Roman')
11 locale.setlocale(locale.LC_ALL, '')
12 figure(figsize=(7, 2.2))
13 grid(color='DarkTurquoise', alpha=0.75, linestyle=':', linewidth=0.5)
14
15 plot(x, f(x), lw=1)
16 for x1, x2 in list(zip(x1k, x2k)):
17     plot([x1, x2], f(np.array([x1, x2])), 'or', ms=5, mec='k')
18     part_x = np.linspace(x1, x2, 1000)
19     fill_between(part_x, f(part_x), y2=a, color='orange', alpha=0.2)
20     hlines(y=a, xmin=x1, xmax=x2, color='b', lw=1)
21
22 gca().xaxis.set_major_locator(MultipleLocator(0.05)), xlim(0, x2k[0] + 0.01)
23 savefig('plot_1.pdf', bbox_inches='tight')
24 gca().xaxis.set_major_locator(MultipleLocator(0.005)), xlim(0.007, x2k[1] + 0.002)
25 savefig('plot_2.pdf', bbox_inches='tight')
```

# Пример использования matplotlib



# Пример пошагового интегрирования сумры

## Step-by-Step Integral Evaluation

```
1 x, y, z = symbols('x,y,z')
2 f = Symbol('f(x,y,z)')
3
4 # Define limits of integration
5 x_llim = 0
6 x_ulim = 2
7 y_llim = 0
8 y_ulim = 3
9 z_llim = 0
10 z_ulim = 4
11
12 print(r'\begin{align*}')
13
14 # Notice how I define f as a symbol, then later as an actual function
15 left = Integral(f, (x, x_llim, x_ulim), (y, y_llim, y_ulim), (z, z_llim, z_ulim))
16 f = x*y + y*sin(z) + cos(x+y)
17 right = Integral(f, (x, x_llim, x_ulim), (y, y_llim, y_ulim), (z, z_llim, z_ulim))
18 print(latex(left) + '&=' + latex(right) + r'\\')
```

# Пример пошагового интегрирования сумры

```
20 # For each step, I move limits from an outer integral to an inner, evaluated
21 # integral until the outer integral is no longer needed
22 right = Integral(Integral(f, (z, z_llim, z_ulim)).doit(), (x, x_llim, x_ulim),
23                 (y, y_llim, y_ulim))
24 print('&=' + latex(right) + r'\\\')
```

25

```
26 right = Integral(Integral(f, (z, z_llim, z_ulim), (y, y_llim, y_ulim)).doit(),
27                 (x, x_llim, x_ulim))
28 print('&=' + latex(right) + r'\\\')
```

29

```
30 right = Integral(f, (z, z_llim, z_ulim), (y, y_llim, y_ulim),
31                 (x, x_llim, x_ulim)).doit()
32 print('&=' + latex(right) + r'\\\')
```

33

```
34 print('&=' + latex(N(right)) + r'\\\')
```

35

```
36 print(r'\end{align*}')
```






# Пример пошагового интегрирования сумру





$$\begin{aligned}\int_0^4 \int_0^3 \int_0^2 f(x, y, z) dx dy dz &= \int_0^4 \int_0^3 \int_0^2 xy + y \sin(z) + \cos(x + y) dx dy dz \\ &= \int_0^3 \int_0^2 4xy - y \cos(4) + y + 4 \cos(x + y) dx dy \\ &= \int_0^2 18x - 4 \sin(x) + 4 \sin(x + 3) - \frac{9}{2} \cos(4) + \frac{9}{2} dx \\ &= 4 \cos(3) + 4 \cos(2) - 4 \cos(5) - 9 \cos(4) + 41 \\ &= 40.1235865133293\end{aligned}$$

Рис.: PythonTex Gallery - [https://github.com/gpoore/pythontex/blob/master/pythontex\\_gallery/pythontex\\_gallery.pdf](https://github.com/gpoore/pythontex/blob/master/pythontex_gallery/pythontex_gallery.pdf)

- Многопоточная генерация билетов;
- Параметр размножения (приоритета) задачи, т.е. сколько вариантов должно содержать данный тип задачи;
- Генерация «преподавательских версий» демовариантов с названиями тем и другими служебными данными в файлах билетов;
- Статистика генерации: сколько каких задач откуда взялось;
- Поддержка русских имен и пробелов в названии файлов с типовыми задачами;
- Расширение набора полезных функций панели управления (например реализация «`compile_folder`»).

# References

-  *Jon Saenz* pygiftgenerator: A python module designed to prepare Moodle-based quizzes, European Physical Society (2020)
-  *Geoffrey M Poore* PythonTeX: Fast Access to Python from within LaTeX, proc. of the 11th Python in Science Conference (SciPy 2012)
-  *Geoffrey M Poore* PythonTeX: reproducible documents with LaTeX, Python, and more, Computational Science & Discovery 8 (2015) 014010
-  *A A B Prasetyanto* Automatic Question Generator System Conceptual Model for Mathematic and Geometry Parallel Question Replication, Journal of Physics: Conference Series 1577 (2020) 012023
-  *Susan EE and Neal MK*, Automatic Question Generation: A More Efficient Process for Developing Mathematics Achievement Items, Journal of Education Measurement (2018) Vol. 55 No. 1 p. 112

-  *Davide P Cervone* MathJax: a JavaScript-based engine for including TeX and MathML in HTML, Joint Mathematics Meetings in San Francisco (2010)
-  SymPy: Python Library For Symbolic Mathematics <https://sympy.org>
-  SciPy 2022, the 21st annual Scientific Computing with Python conference <https://www.scipy2022.scipy.org/>
-  List of Annual SciPy Conferences <https://conference.scipy.org/>

Thank you very much  
for your attention!