分类号_____　　论文选题类型_____

U D C_____　　编号_____

華中師範大學

本科毕业论文（设计）

浅谈我们的我们的大学的大大

大学我们的大学的大大大大

学　　院　数学与统计学学院

专　　业　数学与应用数学(师范)

年　　级　2017级

学生姓名　作者

学　　号　2015123456

指导教师　张三丰教授　三丰教授

二〇二二 年 四 月

华中师范大学

学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师指导下独立进行研究工作所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

学位论文作者签名：　　日期：　　年　月　日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士学位论文评选机构将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于

1、保密　□，在_____年解密后适用本授权书。

2、不保密　□。

（请在以上相应方框内打“√”）

学位论文作者签名：　　日期：　　年　月　日

导师签名：　　日期：　　年　月　日

目 录

内容摘要 1

关键词 1

Title 1

Abstract 1

Keywords 1

1 引言 2

2 LaTeX中制作课件所需文档及常见的命令和环境 2

2.1 制作课件的文档 2

2.2 制作课件的常见命令 3

2.3 制作课件的常见环境 4

3 LaTeX中制作课件所需的常见宏包 4

3.1 不同类型的宏包介绍及其选用 5

4 LaTeX中输入文本、命令和环境的结构排版 6

4.1 制作课件时输入文本、命令和环境的结构排版 6

4.2 LaTeX中制作课件的背景和主题 6

4.3 课件背景主题 7

5 LaTeX中制作课件常用的设计与排版 8

5.1 断行与分页 8

5.2 空行与分栏 9

5.3 目录、标题和小标题 10

5.4 列表的设置 11

5.5 图片的插入 12

5.6 表格的插入 13

5.7 数学公式、定理的生成 14

5.8 图像图形的绘制 16

5.9 课件中的超链接 17

5.10 课件放映的动画效果 18

6 LaTeX中课件制作的实例说明 20

6.1 生成一些特殊符号的说明 20

6.2 高中数学中《幂函数》一节的实例编写说明 21

参考文献 31
致谢 32
附录 33

内容摘要: 摘要的字数控制在400个汉字左右. 尽量不要出现公式参考文献. 介绍本文的主要工作以及创新点.

关键词: 关键词1　　关键词2　　关键词3

Title: Discuss

Abstract: For.

Keywords: keywords1　　keywords2　　keywords3

1 引言

众所周知, 在日常学习和教学中, 好的课件制作方式可以让课件达到耳目一新的效果. 近年来, 使用PowerPoint制作课件在我们的学习和教学中占据了主要的方式, 纵然PowerPoint操作简单, 但千篇一律的PowerPoint课件容易引起人们的审美疲劳, 从而降低了我们的学习和教学的效率. 然而, LaTeX有非常专业的版面设计, 有着非常整洁漂亮的文档排版效果, 使用LaTeX编译出来的文档就如“印刷品”一般, 并且操作简单. 使用者只需学会LaTeX的一些常用逻辑结构的简单易懂的环境及命令, 引用LaTeX系统中相应的宏包, 就可对文档进行自我编译, 在整个过程中不必对实际的文档进行版面设计的修改, 因此也有着高效的排版效率. 此外, 对于很多复杂的版面结构, 如注脚、引用、目录和参考文献等可使用简单的环境命令生成. 特别对于复杂的表格和数学公式的生成, LaTeX的排版表现十分完美, 有着一般排版软件无法媲美的优势. 利用LaTeX制作出的课件, 设计排版新颖、简洁、美观, 内容上章节明了, 可根据不同需要突出重点, 不仅具备了PowerPoint 的诸多功能, 还能展现自身的优势特点所在. 可谓是如今除了使用PowerPoint制作课件的另一个明智之选.

2 LaTeX中制作课件所需文档及常见的命令和环境

在LaTeX系统中, 可生成多种类型的文档. 常见的有article、report、book、beamer和slides. article文档适用于排版本硕论文、科学期刊、演示文稿、短报告等；report文档适用于排版多个章节的长报告、博士论文、短篇书籍等；book文档适用于排版书籍；而beamer文档与slides 文档都是用于幻灯片的排版, 进行课件制作. 在LaTeX程序处理中, 为了使得LaTeX 按照使用者的想法去进行某种操作, 就必须要有相应的一些命令和环境. 在课件制作过程中LaTeX中输入的命令与环境的方式与其他文档的都相同, 这些命令及环境在LaTeX的输入文本中是可见, 但在经过系统的编译之后, 其输出的结果中就不存在了. 为此, 在制作课件的过程中, 认识命令及环境的作用是尤其重要的.

2.1 制作课件的文档

在课件制作的过程中, 所选择的文档主要为beamer, 当然也可选择slides 或者其他类型, 这只需根据使用者的个人喜好即可. 但两种类型的课件制作文档在使用时所需要的宏包种类会有差异, 需要进行相应的调整与改动.

2.2 制作课件的常见命令

LaTeX中大多数的命令语法是由\(反斜杠)后加LaTeX语言程序中所能编译的要求, 再接一个[]与一个或者多个{},[] 中输入的是要求的可省限制条件, {}中输入的是要求中的不可省限制条件. 以下则为开始制作课件时所需的一些常见命令：

- 定义文档类型：\documentclass[CJK,notheorems,compress]{beamer}
- 使用宏包命令：\usepackage{文档编辑所需宏包}, 例如支持中日韩文输出的宏包与设置行间距的宏包分别为：

  \usepackage{CJK}、\usepackage{setspace}
- 使用主题命令：\usetheme{课件主题}, 例如课件的Warsaw 主题与CambridgeUS分别为：

  \usetheme{Warsaw}、\usetheme{CambridgeUS}
- 设置课件版式命令：\setbeamertemplate{\footline}[frame number]
- 设置幻灯片之间的覆盖方式：\setbeamercovered{覆盖方式}, 例如幻灯片之间透明的覆盖方式如下：

  \setbeamercovered{transparent}
- 设置首行缩进2字符：\setlength{\parindent}{2em}
- 文本字体大小、加粗和颜色设置的命令分别为：

  字体大小：{\fontsize{字体尺寸}{行距}{需要设置的文本}}. 此外, 还可以用命令\tiny(\small 或\large或\LARGE或\huge或\Huge){文本} 来调整；

  字体加粗：{\bf{需要设置的文本}}；

  字体颜色：\textcolor{颜色+!+数值(不宜超过200)}{需要设置的文本}；

  例如要将l like Latex! 设置为红色加粗40点大小的文本则可编写为：

  {\fontsize{40pt}{24pt}{\ bf{\textcolor{red!80}{l like Latex!}}}}

  或者可写为：{\bf{\textcolor{red!80}{\fontsize{40pt}{24pt}{l like Latex!}}}}.

若有多个不可省限制条件时, 不可省限制条件中的每一个因素都必须位于一个大括号内, 而且顺序要同命令语法中的顺序相同, 不能随意填入, 也不能交换填入. 例如要输出一个指定宽度和高度的黑色矩形, 输入的语言为：\rule[提升高度]{宽度}{高度}, 以此一个宽10cm, 高5cm的矩形可以输入的语法为：\rule{10cm}{5cm}, 此处没有给出可省限制条件.

2.3 制作课件的常见环境

在LaTeX中, 一个环境是用命令\begin{环境}来作为文本输入的开始, 用命令\end{环境}来作为一段文本输入的结束. 在制作课件时, 如果我们选用课件制作文档, 即是\documentclass{beamer}想要在一张幻灯片上输入相应的文本, 则至少需要用到两个环境, 进行如下排序编写：

\begin{document}

\begin{frame}

文本内容

\end{frame}

\end{document}

即在命令\begin{document}, \begin{frame}之后开始文本输入, 在文本输入结束时, 加上\end{document}, \end{frame}即可. 从上述可知, 在Latex语言输入时, 环境是可以嵌套的, 并且可以进行多层嵌套, 但要保持好调用时的次序, 不能交错. 其嵌套模式为：

\begin{环境1}

\begin{环境2}

文本内容

\end{环境2}

\end{环境1}

不仅如此, LaTeX中很多复杂的排版都可以通过环境得到解决, 在制作课件时也是如此.

此外, 在LaTeX中还有许多常见的环境, 在制作课件的过程中也经常使用到. 例如用于列表的三个环境：itemize、enumerate 和description；分别用于左对齐、右对齐和居中的三个环境：flushleft、flushright 和center；用于表格生成的环境：tabular；用于绘图的环境tikzpicture；由于图片插入的环境：figure；还有用于数学公式生成与方程输出的环境：displaymath 和equation 等等. 而环境的基本语言是：\begin{所需环境}+ 文本+\end{ 所需环境}, 这些环境的具体运用将在下文中LaTeX中制作课件的常用设计排版处给予说明.

3 LaTeX中制作课件所需的常见宏包

在我们制作课件时, 需要用到Latex中大量的宏包. 因为在排版课件中的文档时, 如果没有相应的宏包作为支持, 在LaTeX中我们进行编写的语言想要得到正确的输出就需要更多更复杂的语法代码来支持. 比如说, 如果要在文档中使

环境类型	环境名称
列表环境	itemize、enumerate 、description
对齐方式环境	flushleft、flushright 、center
表格生成环境	tabular
绘图环境	tikzpicture
图片插入环境	figure
数学公式生成 方程输出环境	displaymath、equation
分栏环境	minipage、columns

表 1: 环境

用环境\begin{figure}+需要插入的图片+\end{figure}插入一张图片, 则需要引用宏包\usepackage{graphicx}, 而宏包的引用语法为：

\usepackage[可省限制条件]{所需宏包}

3.1 不同类型的宏包介绍及其选用

经上述我们已经知道了宏包引用的一般语言, 以下介绍制作课件时的一些常见的宏包(参见表1)及使用.

宏包名称	用途
CJK	支持中、日、韩文
CJKfntef	CJK 下的中文下划线、浪线、加点等标记
Latexsym	一些符号
ctexcap	文档类
amssymb, amsmath	各种数学公式、符号
graphicx,graphics	插入图片
tikz, eepic	绘制图形图像
float	浮动体的控制
langtable	分页的表格
makecell	表格中换行
multirow	合并单元格
times	英文字体
color	字体及背景颜色
wallpaper	设置图片背景
hyperref	超链接与其他PDF专有功能
setspace	设置行距
animate, cases,soul	绘制动画
indentfirst	首行缩进

表 2: 常用宏包.

在插入图片和绘制图形图像选用宏包时, 建议读者在插入图片时选用graphicx 宏

包, 绘制图形图像时选用tikz宏包, 因为这两个宏包与相同作用的宏包相比其兼容性要强一些, 使用起来也更加便捷高效.

4 LaTeX中输入文本、命令和环境的结构排版

在LaTeX中输入所要输出的文档时, 需要与命令和环境结合输入, 但命令和环境不会被输出, 只有文本会被输出来. 当所要输出的文档与命令和环境结合输入时, 在结构上需要一定的排版. 制作课件的排版结构与LaTeX中其他文档类的大致相同.

4.1 制作课件时输入文本、命令和环境的结构排版

LaTeX中输入文本、命令和环境的结构排版, 一般分为两个区域, 第一个区域我们称为“导言区”, 第二区域就是我们所谓的正文部分了. 而更明确一点可以以命令\begin{document}来区分, 这个命令之前的区域就是导言区, 而在命令\begin{document}之后与命令\end{document}之前的中间区域就是正文部分了. 在正文部分的输入时, 如果想要达到编辑者理想中的效果, 那就还需要在其中搭配其他相应的命令和环境. 同样, 制作课件时也是如此. 在导言区的第一行, 需要用\documentclass[可省限制条件]{文档类型}这个命令说明所写文档的类型. 从第二行开始, 需要说明文档引用的宏包；对文本的排版, 其中包括页面尺寸、页边距、字距、行距、段落间距、首行缩进、页码等. 用下图以制作课件的一个简单示例进行说明. 另外, 此模板中%后的文字为文字下面的说明, 若编写在LaTeX中, 因为%的存在, 是不会作为文本被输出的.

4.2 LaTeX中制作课件的背景和主题

为了使得我们制作的课件看起来不单调, 我们可以对课件里的幻灯片的背景和主题进行添加和调整, 这样整个课件就会变得绚丽多彩, 具真正的可观性.

```
\documentclass[CJK]{beamer}
\usepackage{ctex}
\usepackage{enumerate}
\usepackage{amssymb,amsmath}
\usepackage{graphicx}
\usepackage{algorithm,algorithmic}...
%设置背景
\mode<presentation> {\setbeamertemplate{background canvas} [vertical shading][bottom=red!10, top= blue!10]}
%设置block的特征
\setbeamertemplate { blocks} [rounded][shadow = true]
%设置主题
\usetheme{Warsaw}
%设置覆盖的效果
```

```
\setbeamercovered{transparent}
\usefonttheme[onlysmall]structurebold
%设置数学公式的字体
\usefonttheme[onlymath]{serif}
\setlength{\parindent}{2em}%首 行缩进
%以上为导言区, 以下为正文部分
\begin{document}
%第一张幻灯片
\AtBeginSection[目录标题]
{\begin{frame} <beamer>
\tableofcontents[currentsection]
\end{frame}}
\ title[幻灯片标题]\\huge 强调}
\author[作者]{邮箱\\{\small{\href{mailto: xxx@mail.com}{xxx@mail.com}}}
\institute[作者所在单位]
{作者所在科室}
\date{\today(当天日期)或者指定日期}
\begin{frame}
\titlepage
\end{frame}
% 目录(第二张幻灯片)
\begin{frame}[allowframebreaks]
\tableofcontents
\end{frame}
\section{章节概述}
%第三张幻灯片
\begin{frame}
\frametitle{标题}
正文编写(需要在幻灯片上呈现的内容)
\end{frame}
%第四张幻灯片
\begin{frame}
\begin{center}
\huge Thanks for your attention!(致谢等)
\end{center}
\end{frame}...
\end{document}
```

4.3 课件背景主题

通常在我们设置课件背景时, 一般有两种方式. 第一种就是改变原来背景的颜色搭配模式, 产生不一样的搭配方式, 来达到我们的目的, 使用的命令语法如下：\mode<presentation> {\setbeamertemplate{背景填充方式}[填充效果][颜色搭配情况]} 例如可编写为：\mode<presentation> {\setbeamertemplate{background canvas}[vertical shading][bottom=red!15,top=blue!20]}

此处我们可以发现这个命令的语言和一般命令的语法不一样, 这是我们值得注意的地方. 第二种就是插入背景图片, 以一张图片来作为课件中幻灯片的背景, 这种方式需要我们在导言区引入宏包wallpaper才能将我们准备好图片设置为课件中幻灯片的背景, 这种设置方式的语法为：

\setbeamertemplate{背景填充方式}{\TileWallPaper{宽度}{高度}{文件名}}

如, \setbeamertemplate{background canvas}{\TileWallPaper{60pt}{45pt}{图片1}}.

在此种方式中我们可以调整插入图片的宽度和高度来完成有效的背景设置, 如果该条命令是放在导言区中, 则课件中的每张幻灯片都是以插入的这张图片为背景. 但我们也可以将这条命令放入正文部分的每张幻灯片的上方, 更换掉插入的图片, 就可以使得课件中每张幻灯片的背景图片不一样. 此外, 如果觉得背景图片过于单调, 我们也可以用多个\TileWallPaper命令将多张图片连接起来使用, 即如下例所示：

\setbeamertemplate{background canvas}{\TileWallPaper{宽度1}{高度1}{图片1}
\TileWallPaper{宽度2}{高度2}{图片2}\TileWallPaper{宽度3}{高度3}{图片3}}

除了设置背景之外, 我们常常用到的就是主题设置了. 而主题的设置比较简单, beamer提供了不同种类风格的幻灯片主题, 只需在导言区使用\usetheme{主题名称}命令选择. 其中已定义的主题有Bergen、Berkeley、Berlin、Boadilla、CambridgeUS、Singapore、Szeged、Warsaw等等. 事实上, beamer的主题是由内部主题（inner theme）、外部主题（outer theme）、字体主题（font theme）、色彩主题（color）等组合而成的, 以上四个不同类型的主题又可以作为一个独立的部分进行更改, 分别使用\useinnertheme{主题名称}、\useoutertheme{主题名称}、\usefonttheme{ 主题名称}、\usecolortheme{主题名称}进行选择. 例如想更换一个主题的色彩主题, 则使用\usecolortheme{色彩主题名称}这个命令, 将其放入导言区即可. 而四个类型的主题在beamer中已有大量的定义, 使用时只需进行查询, 然后运用. 已定义的内部主题有circles、rectangles、rounded等等；外部主题有sidebar、split、shadow、tree等等；色彩主题有rose、seagull、seahorse等等；字体主题有serif、structurebold、structureitalicserif等等.

5 LaTeX中制作课件常用的设计与排版

在LaTeX中制作课件时, 想要在一些排版设计上达到相应的理想效果, 就需要调用相应的命令和环境.

5.1 断行与分页

在制作课件的过程中, 如果我们需要用到断行, 则只需要用到如下命令就可以了.

\\或\newline

如果在英文输入时某些单词使得段落不整齐, 那么就需要将这些英文单词断开, 那就需要断词, 也就是说将某个英文单词断开, 那就只需要用到命令\- 即可, 例如要将interesting 这个英文单词在t与i之间断开, 那么只需在LaTeX中将单词这样输

入interest\-ing即可.

对于LaTeX中的其他文档编写而言, 有时候可能需要另起一页, 也就是分页. 这个时候我们需要用到的就是\newpage 这个命令了. 但在制作课件的过程中, 由于我们的幻灯片本来就是一页一页的, 所以自然就不需要进行分页了, 只需要控制一页幻灯片上的输入内容量, 使其合理美观即可. 当然, 如果要另起一张新的幻灯片, 那么只需用到frame环境, 以命令\begin{frame} 开始, 再以命令\end{frame}结束就行了, 若是要在幻灯片上呈现文本, 则将想要输出的文本输入到两条命令之间即可. 若要在幻灯片上呈现一些复杂的内容, 则将文本和环境结合起来输入到两条命令之间就可以完成了. 与此同时, 若是想要几张幻灯片, 那么使用几个frame 环境就可以了.

5.2 空行与分栏

当我们设置一张幻灯片上的内容结构时, 为了排版上的美观和内容上的充实, 我们需要将上下两段内容空开一行或空开多行或自定义行距, 这时我们就可以用如下的两个命令来解决.

空一行：\bigskip,　自定义行距：\vspace*{距离}或\vspace{距离}

例如要将上下两段的行距设置为30pt或者2cm, 则其命令可写为：\vspace{30pt}或者\vspace{2cm}. 若需要将同一行中的文本间隔一段距离, 则可用两种方法加以执行, 第一种是在想要断开的文本处加上一个{}即可, 但此种方式的间隔距离不大, 有时候达不到想要的效果；第二种则是使用命令：

横向间距：\hspace{距离}

其使用方法也是将其放入想要断开的文本处即可. 比如说在编辑幻灯片标题“第2章趣味数学”时, 在LaTeX中可编辑为“第2章\hspace{20pt} 趣味数学”, 这样在输出的文档中“第二章”后面“趣味数学”前面将会有一段间距.

制作课件时, 如果想要在一张幻灯片的左边部分编辑文字, 右边部分插入图片或者其他, 亦或是将其结构反过来. 这个时候, 我们就需要将幻灯片分为左右两个部分或者多个部分, 使得一张幻灯片成为多栏结构. 而此时使用multicols环境即可, 它可以分为多栏, 其简要的语法如下：

- \begin{multicols}{2}
- 文本或图片或其他内容
- \end{multicols}

以上语法仅是将一张幻灯片分为两栏结构的简单写法, 若想增加分栏数, 将数字2变成别的数字即可. 另外, 在需要使用分栏结构时, 也可使用环境\begin{columns}, 可

将页面分为两栏或者多栏, 多栏结构利用多个column环境即可. 其两栏的基本语法如下：

```
1. \begin{columns}
2. \begin{column}{a\textwidth}
3. 文本或图片或其他内容1
4. \end{column}
5. \begin{column}{b\textwidth}
6. 文本或图片或其他内容2
7. \end{column}
8. \end{columns}
```

此处值得说明的是, 环境column中每栏所占比例之和可以大于1.

5.3 目录、标题和小标题

目录一般处于课件中的第二张幻灯片上, 设置目录可以使用命令\tableofcontents来插入到想要放入的某张幻灯片上. 每一个小结的目录标题可以使用\section{章的标题1}命令来设置. 其基本语法可如下所示：

```
1 \section{第一章　章的标题1}
2 \section{第二章　章的标题2}
3 \section{第三章　章的标题3}
4 %目录内容会呈现在下面这张幻灯片上
5 \begin{frame}
6 \tableofcontents
7 \end{frame}
```

此处需要说明的一点是, \section{章的标题1}可以写在环境frame 之外, 也可以写在环境frame之内的\tableofcontents命令下面. 此外, 若是文本中的章后还有节和小节, 则只需在命令\section{章的标题1}后接连使用命令\subsection{节的标题1}与命令\subsubsection{小节的标题1}即可.

在课件中, 一般每张幻灯片上都会有一个标题乃至一个小标题, 若是想在一张幻灯片上设置标题, 其设置方法有两种, 则对应的语法及简单语言分别如下：

```
1 第一种设置方式：
2 \begin{frame}
3 {标题}{ 小标题}
4 文本或其他内容内容
5 \end{frame}
6 第二种设置方式：
7 \begin{frame}
8 \frametitle{标题} \framesubtitle{小标题}
9 文本或其他内容内容
10 \end{frame}
```

第二种设置方式比第一种设置方式要复杂一些, 但是两者的效果是相同的, 可以得出相同的结果. 但当我们使用第一种设置方式时, 由于LaTeX系统中的某些兼容性问题, 有可能会出现一些预想不到的结果, 这时我们换用第二种方式即可.

5.4 列表的设置

多列同等地位的条目出现在幻灯片中是很常见的, 将这些条目从上到下的排列我们简称列表. 对于列表的设置, 有三个环境可以使用, 并且三个环境之间可以进行相互嵌套使用. 这三个环境分别为：Itemize、Enumerate 和Description, 三个环境使用效果基本上相同, 仅是列表中每个条目前面的标签不一样, 使用者可根据其需求进行相应类型的选择. 列表环境的使用基本语法为：

```
1 \begin{环境名称}
2 \item{条目内容一}
3 \item{条目内容二}
4 \item{条目内容三}
5 ............
6 \end{环境名称}
```

如下是一个列表嵌套的示例.

原代码

```
院系设置以及专业分配
1 \begin{itemize}
2 \item{数学与统计学学院}
3 \begin{enumerate}
4 \item 数学与应用数学专业
5 \begin{description}
6 \item 免费师范生
7 \item 非免费师范生
8 \end{description}
9 \item 统计学专业
10 \item 其他专业
11 \end{enumerate}
12 \item{文学院}
13 \begin{description}
14 \item 汉语言文学专业
15 \item 其他专业
16 \end{description}
17 \item{外国语学院}
18 \item{............}
19 \end{itemize}
```

显示效果

院系设置以及专业分配

- 数学与统计学学院
  1. 数学与应用数学专业

     免费师范生

     非免费师范生
  2. 统计学专业
  3. 其他专业
- 文学院

  汉语言文学专业

  其他专业
- 外国语学院
- ....

以上列表条目前的标签是可以自己定义的, 在自行定义之前需要在导言区使用宏包命令\usepackage{enumerate}, 用于识别item 中的序号（1)、(2）等等, 还需要定义

命令\renewcommand{\labelitemi}{\bullet}, 用于识别item中的-、*和大圆点等符号.

对于itemize环境里标签的形式, LATEX中的简单列表默认生成的是一个小圆点, 若想要更换一些不一样的列表标签, 比如-、* 等这样的. 则其语法为：

输入代码	显示效果:
1 \begin{itemize}	- I like Latex!
2 \item[-] I like Latex!	* I like to use Latex!
3 \item[*] I like to use Latex!	
4 \end{itemize}	

对于enumerate环境而言, 标签是带序号的列表, LATEX系统默认生成的是1、2、3 等自然数. 如果想要更换为其他的, 比如带小括号的序号标签(1)、(2)、(3)等或A、B、C等. 其使用语法如下：

输入代码	显示效果:
1 \begin{enumerate}[(1)]	(1) I like Latex!
2 \item I like Latex!	(2) I like to use Latex!
3 \item I like to use Latex!	
4 \end{enumerate}	

对于列表环境中标签的设置, description是最为简单的, 该环境LaTeX 系统中默认的标签是没有的, 若想为其设置标签, 其使用的语法基本上和itemize 环境的相似, 在命令\item后使用[](中括号), 在中括号中填入需要的标签就可以了.

5.5 图片的插入

若是想在课件中的某张幻灯片中插入一张图片, 那么需要在文档的导言区使用插入宏包命令\usepackage{graphicx}才可以. 并且要将需要插入的图片文件与tex文件放入同一个文件夹下, 否则就需要调整相应的路径才能得以实现. 插入图片的基本语法如下：

```
1 \begin{figure}
2 \centering    %(表示图片居中放置)
3 \includegraphics[可省限制条件]{图片名称(需要带上图片的格式)}
4 \caption{图片标题}
5 \end{figure}
```

在上述语法中, 可省限制条件包括文本中插入图片宽度、高度、放缩程度、旋转

角度等. 宽度可以用“width=数值\textwidth”(此处的数值一般小于等于1, 表示插入图片的宽度是文档宽度的多少倍)和“width=数值pt”(此处的数值插入图片的宽度是多少点)来说明；高度一般可以用“height=数值\textheight”(此处的数值一般小于等于1, 表示插入图片的高度是文档高度的多少倍)和“height=数值pt”(此处的数值插入图片的高度是多少点)来说明；放缩程度可以用“scale=数值”(此处数值表示将插入图片放大或者缩小到原图片的多少倍)来说明；旋转角度可以用“angle=数值”(此处数值表示将插入图片逆时针旋转多少度)来说明. 若是想同时有几个限制条件, 则将其编辑到[] 中用英文逗号隔开即可. 并且在编辑图片名称时, 一定要注意带上图片的文件格式, 一般都为png与jpg格式, 例如要插入图片名为aaa, 格式为png的图片, 则上述语法中第四条命令的{}中输入的内容为aaa.png. 如果在插入图片时, 图片的排版没有插入到预定的位置, 而是放到了文档的最前面, 这时可以使用\usepackage{float}, 将其放到导言区, 然后再将命令\begin{figure}换为“\begin{figure}[H]”即可.

另外, beamer中除了有标准的插入环境, 还提供了一个插入较小图标以便放在幻灯片边角的命令\logo. \logo命令放置的位置一般为导言区, 并且可用这个命令来添加幻灯片上的校徽、商标等图片. 例如将一张图片名为abc, 格式为png的图片设置在幻灯片上, 那么其命令为：\logo{\includegraphic{abc.png}}

5.6 表格的插入

在幻灯片上插入一些表格是特别常见的, 例如要插入一个6行4列的表格, 那么它的基本语法为：

```
\begin{tabular}{r|cc|c} \hline
姓名&语文&数学&英语\\ \hline
小王& 65&88&75\\
小李&90&84&78\\ \hline
小张&85&68&80\\ \hline
小石& 64&77&89\\
小陈&91&87&91\\ \hline
\end{tabular}
```

显示效果

姓名	语文	数学	英语
小王	65	88	75
小李	90	84	78
小张	85	68	80
小石	64	77	89
小陈	91	87	91

由以上编写语言和输出表格的效果可以看出, 语言编写过程中是从表格的第一行开始从左到右的编辑, 第一行写完之后再转行写第二行, 以此类推, 每行中每两个相邻的单元格用符号“&”隔开, 每行写完后都要加上转行符号“\\”, 若需要横线, 则在其后加上命令\hline, 若不需要, 则可不加. 在{r|cc|c}中, 每个字母代表一列, 并且“r”代表表格该列的文本是右对齐, “c”代表表格该列的文本是位置居中, 如

果想设置为左对齐, 则用“l”即可. 当表格中不需要某两列之间的竖线时, 只需去掉{r|cc|c}中指定的两列间的“|”即可. 此外, 若需要将表格整体向左右移动, 则在命令\begin{tabular}{r|cc|c}前使用命令\hspace{移动距离}即可. 移动距离为负, 则向左移；移动距离为正, 则向右移.

在生成表格的过程中, 合并表格中的部分单元格是插入表格的重要内容. 表格单元格的合并分为“列的合并”与“行的合并”. 对于列的合并, 在表格编写的过程中使用命令\multicolumn{合并的单元格个数}{合并后表中列的分布}{合并后单元格中的文本内容}就可以将某行中的相邻的几个列合并为一个单元格. 对于行的合并, 稍微要简单一些, 可以使用画线命令\cline{某列-某列}来合并, 比如从要从第二列至第五列有横线, 那么可写为\cline{2-5}, 即是给出画线的起始位置和终止位置即可. 在生成表格时, 命令\hline是画出一条完整的横线, 命令\cline是画一条不完整的横线. 但这样我们会发现在几行合并后的单元格中的文本内容没有居中的效果, 从而有时候会影响了表格的美观程度, 这个时候我们可以在导言区引用宏包, 即使用宏包命令\usepackage{multirow}, 同样在表格编写的过程中使用命令\multirow{合并的单元格个数}∗{合并后单元格中的文本内容}. 如下给出示例：

代码

```
1 \begin{tabular}{|c|r|r|r|}\hline
2 \multirow{2}*{姓名}&\multicolumn{2}{c|}{成
    绩}\\
3 \cline{2-4}        &语  文&数  学&  英
    语\\\hline
4 小王& 85 & 95 & 88\\ \hline
5 \end{tabular}
```

输出效果

姓名	成绩		
	语文	数学	英语
小王	85	95	88

5.7 数学公式、定理的生成

对于数学公式的生成, 在设置数学公式排版时, 有时候为了排版上的美观, 进行设置的方式也不同. 可以使用的环境有equation、split、displaymath等. 各个环境的使用方式以以下例子进行说明.

例如将公式$a^x + a^y \neq a^{x+y}$输入到幻灯片上, 其语言及效果如下：

输入代码	输出效果
\begin{equation} a^x+a^y\ne a^{x+y} \end{equation}	$a^x + a^y \neq a^{x+y}$

在带有多个等号的等式中, 为了美观, 我们常常需要将等号对齐排版, 如：

输入代码	输出效果
\begin{equation*} \begin{split} z&=(a+b)^2 \\ &=a^2+2ab+b^2\\ &=b^2+2ab+a^2\\ &=a^2+b^2+2ab\\ \end{split}\\ \end{equation*}	$$\begin{split} z &= (a+b)^2 \\ &= a^2+2ab+b^2 \\ &= b^2+2ab+a^2 \\ &= a^2+b^2+2ab \end{split}$$

在上述例子中, 值得注意的是split环境需要equation环境嵌套在外层才能够得出结果, 不然将会出错.

另外, 有些公式或者方程需要插入到某一句话内, 或者说插入到某行之中, 那只需要将想要插入的公式或者方程在两边各自加上一个“$”符号即可. 还有一种就是要将公式或者方程放入两行文本之间, 第一种方法我们可以使用上述三个插入数学公式的环境即可做到；第二种方法是在公式或者方程的两边各自加上“$$”即可. 示例如下：

插入情况	输入代码	实际效果
行内插入	$ \lim_{n\to\infty} \sum_{k=1}^n\frac{1}{k^2} =\frac{\pi^2}{6}$	$\lim_{n\to\infty}\sum_{k=1}^n\frac{1}{k^2}=\frac{\pi^2}{6}$
行间插入	\begin{displaymath} \lim_{n\to\infty} \sum_{k=1}^n\frac{1}{k^2} =\frac{\pi^2}{6} \end{displaymath}	$$\lim_{n\to\infty}\sum_{k=1}^n\frac{1}{k^2}=\frac{\pi^2}{6}$$

对于定理的设置, 在beamer中已定义了许多相关环境, 例如theorem、corollary、definitions、example和block等. 大多数定理环境中, “定理”和“定义”都以英文名给出, 例如theorem环境中定理前面的名称就是“Theorem”. 此外, 若是需要将定理定义的英文名换成中文, 那么使用命令\newtheorem{原定理英文名称}{新定义名称}, 例如\newtheorem{theorem}{定理}. 然而, block环境要特殊一些, 它是形成一个小板来显示突出定理的存在, 可以大大增强幻灯片的美观性. 将定理的环境运用的语言示例如下：

```
\begin{frame}
\begin{block}{正弦定理}
在一个三角形中, 各边和它所对的正弦
    的比相等, 即:
\begin{equation}
\frac{a}{sinA}=\frac {b}{sinB}=\frac{c}{sinC}
\end{equation}
\end{block}
文本或其他内容
\end{frame}
```

5.8 图像图形的绘制

对于图像图形的绘制, 需要在导言区引入宏包, 即使用宏包命令\usepackage{tikz}, tikz绘图宏包是LATEX中最为强大的绘图宏包. 除了宏包的使用之外, 还需要使用tikzpicture

环境以及\draw等命令. 现将一个简单的示例如下:

```
\begin{tikzpicture}[scale=1.2]
\draw[->](-1.5,0)−−(1.5,0)node[left,below, font=\tiny]{$x$};
\draw[->](0,-0.2)−−(0,2.5)node[right, font=\tiny]{$y$};
\foreach\x in{-1,0,1}{\draw(\x,0)−−(\x,0.05)node[below, outersep=2pt,font=\tiny]at(\x,0){\x};}
\foreach\y in{1,2}{\draw(0,\y)−−(0.05,\y)node[left,outer sep=2pt,font=\tiny]at(0,\y){\y};}
\draw[color=red, thick,smooth,domain=-1.4:1.4]plot(\x,{\x∗\x}) node[left]{$y=x^2$};
\end{tikzpicture}
```

以上语法的输入所画图像如下:

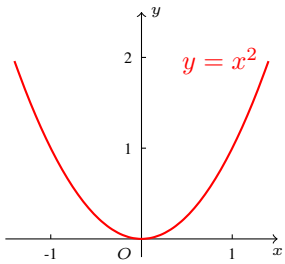

图 1: 二次函数

在上述图像绘制的示例中, tikzpicture是绘图环境, draw是绘图命令, “[scale=1.2]”代表将绘制出来的图形放缩为所绘图形的1.2 倍, 此处与插入图片的类似, 在[]中除了放缩大小外还可填入高度、宽度和旋转角度等条件. 前两个\draw命令是绘制x轴与y轴, “[->]”表示绘制x轴与y轴的箭头, (-1.5,0)−−(1.5,0)与(0,-0.2)−−(0,2.5)分别表示绘制x 轴与y 轴的线条及长度, 例如x轴即是绘制点(-1.5,0)到点(1.5,0)这一段; 并且整个语法中的“node[]{}”都是对前面所绘图形进行命名, []中可填入所命名称的放置位置, 包括left、right、above和below, 还可填入所命名称的字体大小颜色

等限制条件, 例如node[right,font=\tiny]{x}代表将x放入x轴的右边, x为最小号. 后两个\foreach都是对x轴和y轴进行取点, 画刻度线, 例如“{-1,0,1}”表示在x轴上取出-1、0、1三个点. 最后一个\draw 命令是绘制二次函数$y = x^2$的图像, “[color=red, thick,smooth,domain=-1.4:1.4]”表示图像线条为红色, 厚实光滑, 定义域为-1.4到1.4. plot(\x,{\x*\x}) 表示画以x为变量的x^2的图像. 值得注意的是, 在绘制函数时, 变量前面必须加\, 相同的变量相乘时必须要以*连接.

5.9 课件中的超链接

课件中使用超链接, 同样要在导言区引用宏包, 即使用宏包命令\usepackage{hyperref}. 在beamer中, 为超链接提供了按钮, 如果想要从某张幻灯片超链接到另外一张幻灯片中, 或是从某个地方链接到另一个地方, 并且还可以链接回来, 那么可以使用命令\bgoto{超链接的幻灯片名称}{按钮名称}和命令\breturn{超链接的幻灯片名称}{按钮名称}, 这两个命令会在幻灯片中形成两个按钮. 在使用以上两个命令的过程中, 要给两张需要超链接的幻灯片各自设置一个标签, 设置时在命令\begin{frame}后加上“[label= 幻灯片名称]”就可以了. 此处值得注意的是在设置标签定义幻灯片名称时幻灯片名称必须使用英文输入, 不能使用中文. 另外在使用\bgoto和\breturn两个命令设置超链接时, 可将两个命令放入本张幻灯片的任意文本内容位置, 课件制作者可根据幻灯片的美观情况考虑生成按钮的具体放置位置. 现将超链接使用方法示例如下：

```
\begin{frame}[label=mingchengyi]
\frametitle{幻灯片标题}
文本图片或其他内容 1.\bgoto{mingchenger}{ 名称二}
\end{frame}
.....(其他幻灯片)
\begin{frame}[label=mingchenger]
\frametitle{幻灯片标题}
文本图片或其他内容 2.\breturn{mingchengyi}{名称一}
\end{frame}
```

另外一个更为复杂一点的是先使用命令\hypertarget{链接目标名称}{文本}来创造超链接对象, 再用命令\hyperlink{ 链接目标名称}{\beamergotobutton{文本}}来完成超链接. 其使用方式和上述方法类似.

5.10 课件放映的动画效果

课件放映时, 动画效果的出现更容易吸引人的眼球, 这就要求我们在制作课件的过程中为其添加动画效果. 除了动画效果, 背景音乐也是有时候的一个两点之一. 所以在课件制作的过程中使用动画效果和背景音乐是设计者使用的两个常用手段.

在beamer中, 除了使用者绘制动画之外, 一种最为基本的动画效果就是“覆盖”, 利用覆盖就可以让我们幻灯片中的内容依次的显示出来, 例如列表中的几个条目, 我们就可以设置动画效果使其一条一条的逐步显示出来. 覆盖效果的设置有多种方式, 以下介绍几个常用的方式.

最常用的方式就是使用命令\pause, 该命令后面的内容将会在该命令前面的内容显示完了之后再显示出来. 例如要逐条显示一个列表中的条目, 只需在每个\item命令前加上命令\pause就可以了. 此处值得说明的一个小技巧是目录的逐条显示, 在上文中已说明过目录的生成方式, 若是想要目录在幻灯片放映时逐条显示出来, 除了在命令\section{内容}前使用命令\pause之外, 还可以将目录命令\tableofcontents换成命令\tableofcontents[pausesections].

其次可以使用命令\onslide<文本内容显示情况> {文本内容}, 如下有源代码

```
\begin{frame}{幻灯片标题}
文本内容 1
\onslide<1> {文本内容 2}
\onslide<2-> {文本内容 3}
\onslide<1,3> {文本内容 4}
文本内容 5
\end{frame}
```

生成幻灯片中的动画效果为：文本内容 1 和文本内容 6会随着本张幻灯片的出现一直显示；文本内容 2和文本内容 4 也会随着该张幻灯片的出现显示出来, 文本内容 3不会被显示, 但之后文本内容 2和文本内容 4会在文本内容 3显示出来的时候被覆盖, 也即是会消失, 而文本内容 3显示出来之后就不会再被覆盖；最后文本内容 4又会被显示出来. 换句话说, 本张幻灯片放映时, 第一个\onslide命令后的内容只在第一步时显示, 第二个\onslide命令后的内容在第二步后会一直显示出来, 第三个\onslide命令后的内容会在第一、三步的时候显示. 再者, 与命令\onslide用法类似的命令还有\only、\uncover和\visible 等, 其效果也具有相似性, 其原理也只是覆盖文字的方式不同而已, \only只在指定的帧占据位置并显示, 并且在播放幻灯片时整个页面会有轻微的抖动, 而\uncover在一组幻灯片里的每一张都占有位置. 此外, \onslide命令与环

境overprint使用时与它单独作为一个命令使用时其效果是不同的.

最后还有一种更为简单的设置方法, 就是使用<>(尖括号)加显示步骤的方法来对动画效果进行覆盖设置. 在beamer中, 很多命令和环境的后面都可以使用<>(尖括号)来完成动画效果的设置, 达到相应的效果, 此种设置方式在列表环境中尤为常见, 并且在列表中使用<>(尖括号)还有一些简洁的使用方式. 对应的几种语言编写和说明示例如下表：

<table>
<tr><td>编写方式</td><td>\begin{itemize}
\item<1->文本内容1
\item<2->文本内容2
\item<3->文本内容3
\end{itemize}</td><td>\begin{itemize}[<+->]
\item文本内容1
\item文本内容2
\item文本内容3
\end{itemize}</td><td>\begin{itemize}[<+->]
\item\textcolor<1,3>
{red!60}{文本内容1}
\item \textbf<2-4> {文本内容2}
\item 文本内容3
\item 文本内容4
\end{itemize}</td></tr>
<tr><td>显示说明</td><td colspan="2">每个\item命令后的内容依次显示</td><td>文本内容逐条显示
但文本内容1在第一步、
第三步变红, 文本内容2
在第二步到第四步加粗</td></tr>
</table>

此外, 还有一种起重点突出作用的逐条显示效果, 是使用命令\item<+-|alert@+>在列表环境中来执行. 其语法如下：

```
\begin{itemize}[<+-| alert@+>]
\item 文本内容 1
\item 文本内容 2
\end{itemize}
```

其次, 课件中还可插入动画图片, 或者自行绘制动画图形, 这需要在导言区引用\usepackage{graphicx}宏包与\usepackage{animate}宏包, 来达到某些需要的动画效果. 对于动画图片的插入, 需要插入将pdf文件的gif图像分成不同帧的png图像, 然后输入如下命令即可：

```
\animategraphics[autoplay,controls,width=0.6\textwidth, height=0.6\textheight]{3}{ra-}{0}{10}
```

在上述命令中, []中填入的是可省限制条件, 第一个{}到第四个{}分别代表的是每秒播放几张图片, 播放png图片的方式, 第一张png图片序号和最后一张png图片的序

号. 例如上述命令中的各个条件所代表的意思为：{controls} 代表文档中会添加手动控制按钮, {3}代表每秒播放3 张图片, {ra-} 代表循环播放的png图片, {0}和{10}代表播放的第一张png 图片和最后一张png图片的序号. 另外, 最后生成的PDFLaTex 需要使用adobe reader等阅读器的支持才能够正常的播放出来.

若是自行绘制动画图形, 或动画图像, 需要在导言区引用\usepackage{tikz}宏包与\usepackage{animate}宏包, 并且还需用到tikzpicture与animateinline两个环境. 例如绘制一个正弦函数的动态图像, 其语法如下：

```
\newcommand{\plotsin}[1]{% \draw[red] plot[domain=#1:#2] (\x,{sin(\x r)});}
绘制正弦函数的动画:\\
\begin{animateinline}[autoplay,palindrome,
begin={\begin{tikzpicture}[scale=1,line width=1.5pt]
\draw[line width=0.5pt,step=0.5,dashed,use as bounding box] (-4,-1.5)grid(4,1.5);
\draw[->,blue] (-4,0)−−(4,0) node[below left]{$x$};
\draw[->,blue] (0,-1.5)−−(0,1.5)node[below left]{$y$};},
end={\end{tikzpicture}}]{5} \multiframe{20}{Rxa=-0.2+-0.2, Rxb=0.2+0.2}{% %Rxa=-0.2,-0.4,...,-4.0 %Rxb=+0.2,+0.4,...,+4.0 \plotsin{\Rxa}{\Rxb}%}%
\end{animateinline}
```

以上为两个参数的正弦函数动态图像, 所生成的画面总共有20帧, 并且以一秒5 帧的速度进行播放.

最后幻灯片中背景音乐的插入, 需要使用宏包multimedia, 即要在导言区使用宏包引用命令\usepackage{multimedia}, 再使用命令\sound[选项]按钮名称音乐文件名即可. 可省限制条件“[选项]”中可用的条件有autostart、automute、loop、repeat、channels(1)、samplingrate (44100)、bitspersample (16)等. 例如要插入一个文件名为notify, 文件格式为wav的背景音乐, 幻灯片放映时的按钮名为yinyue, 其语法可写为：\sound[autostart,samplingrate=705000,bitspersample=16, channels=2]{yinyue}{notify.wav}.

6 LaTeX中课件制作的实例说明

6.1 生成一些特殊符号的说明

在LaTeX中有很多字符是系统中的保留字符, 在LaTeX被赋予了特殊的意义, 或不一定存在于正常的字库之中. 如果直接在文本输入的过程中中输入这些特殊的字符, 它们通常不会被直接输出, 而且有时还可能导致在LATEX系统编译的过程中出现错误.

以下符号

% ^ & _ { } \

输入方式是在这些符号前加上反斜线\

\# \ \% \^{} \& _ \{ \} \{}

其他一些特殊符号可以由数学环境中的特殊命令得到. 反斜线\不能通过在其前面加另一个反斜线得到\\, 在上文中也说过, \\是一个用来换行的命令. 如上述\可以输入“\{}”的方式就可以得到解决, 也可以输入“\backslash”就可得到正确的输出.

6.2 高中数学中《幂函数》一节的实例编写说明

高中必修中函数的学习在中学教学中, 由于函数本身具有抽象性, 给学生的学习和教师的教学带来了一定的困难. 而在教学中对函数部分的形象化能够对学生的学习和教师的教学起到一定的优化效果, 从而提高教学的效率. 若使用PowerPoint来设计课件, 只能进行图片的插入, 而无法绘制大多数的图形与图像, 同时复杂的数学公式输入需要借助于其他的软件, 自身难以独立完成. 很多的动态图片及图形也难以进行绘制, 想要形成需要的动画效果, 就需要在教学时搭配其他的动画制作软件才能得以完成. 但利用LaTeX就可以得到美观整洁, 操作排版流程一体化的课件, 从中可以省略掉很多复杂程序或步骤. 现以下以函数部分的教学实例《幂函数》的LaTeX教学课件设计进行说明, 设计分为三个部分, 分别为：教学导入, 教学过程和教学运用, 幻灯片总的为18 张.

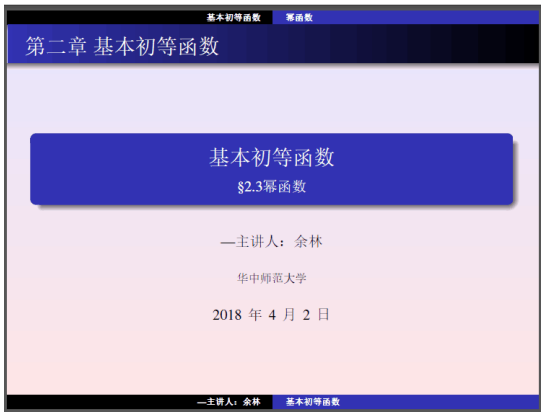

图 1: 第1张幻灯片

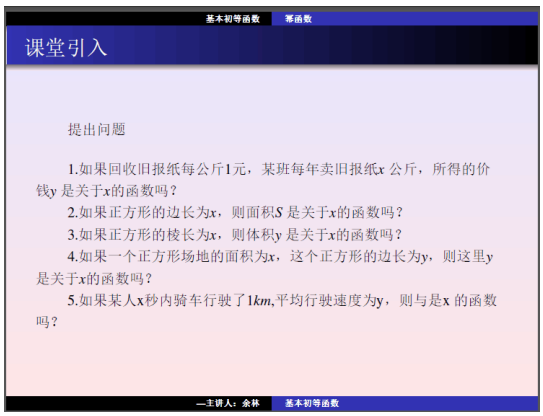

图2: 第2 张幻灯片

第一张幻灯片为封面幻灯片, 上面的内容将会一次性显示出来, 对上课内容的章节和授课人进行了相应说明. 将如下命令\frametitle{第二章{} 基本初等函数}, \subtitle{\§2.3{}幂函数}, \author{—主讲人：余林}, \institute{华中师范大学}, \date{\today}和\titlepage依次放入frame环境中即可. 第二张幻灯片为幂函数的课堂

引入部分, 图片上的五个问题将会依次显示出来的, 以生活中的实例引出学生在前面的学习中所接触过的部分幂函数, 从而引出该节课的新内容. 其效果使用的命令为\uncover.

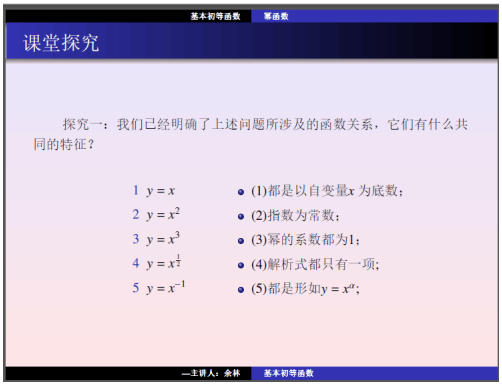

图 3: 第3张幻灯片

第三张幻灯片是通过问题引导学生对幂函数定义的探究, 总结幂函数定义中的共同特征, 对幂函数的定义进行启发和解读. 对于动画效果的呈现, 前5 个常见幂函数会直接被播放出来, 而后面的总结将会依次播放出来.

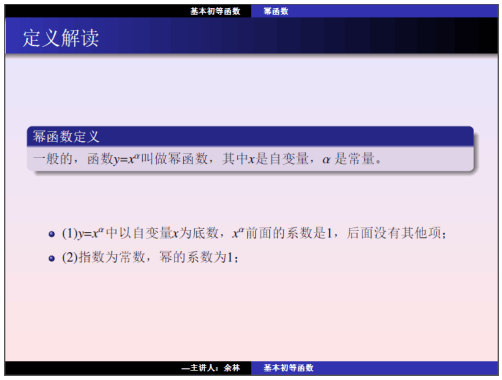

图 4: 第4张幻灯片

第四张幻灯片为幂函数的定义, 此处使用了block环境对定义进行了强调, 使得学生可以对幂函数的定义加强印象, 而定义下面的定义解读使用列表环境即可达到效果, 对学生起重复巩固的作用.

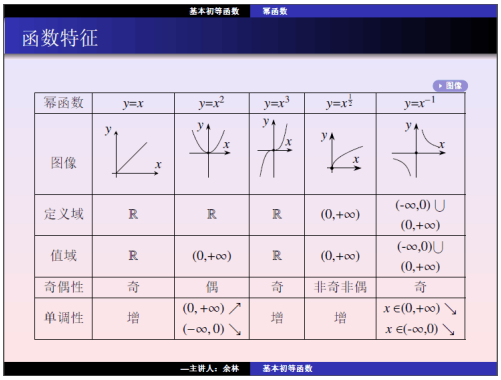

图 5: 第5张幻灯片

第五张幻灯片是对常见幂函数基本特征的复习回顾, 此处可以与学生进行互动, 其表格内的内容是一行一行的显示出来的. 表格中的图像可以直接通过环境tikzpicture绘画出来, 对于表格行的依次显示使用命令\uncover{ 内容}就可达到效果. 并且设有超链接, 按钮“图像”会链接到第六张幻灯片上, 学生先通过对图像的直观观察, 从而得出幂函数的一些基本特征和个别幂函数的内在特征, 从而使学生得到直观结论和更多启发, 再来对表格进行反馈填写. 该张幻灯片的LaTeX语言编写以表格的第二行的第二个单元格和最后一行的最后一个单元格为例：

```
\begin{frame}
\begin{tabular}{r|c|c|c|c|c}
\hline
*&\uncover<2-> {\begin{tikzpicture}[>=stealth]
\draw(0,0)–(.8,.8);
\draw[->](-.1,0)–++(1.2,0)node[above]{$x$};
\draw[->](0,-.1)–++(0,1.2)node[left]{$y$}; \end{tikzpicture}}&*&*&*&* \\ \hline
*&*&*&*&*&*\\ \hline
*&*&*&*&*&*\\ \hline
*&*&*&*&*&*\\ \hline
*&*&*&*&*&*\\ \hline
*&*&*&*&*&\makecell[c]\{$x\in$(0,+$\infty$) $\searrow$
    \\ $x\in$(-$\infty$,0) $\searrow$} \\ \hline
\end{tabular}
\end{frame}
```

以上的LaTeX语言编写, 是给出了该张幻灯片中整个表格框架（6行6列的表格）的编写, 且对表格的第二行的第二个单元格和最后一行的最后一个单元格完成示例编写. 其中上述语言中一个*代表一个单元格的内容, 若是想在相应单元格输入内容,

只需将其替换掉即可；若是不想输入内容, 即是要留一个空白单元格, 将其去掉即可；若是将其保留, 则LaTeX编译的时候将会出现错误.

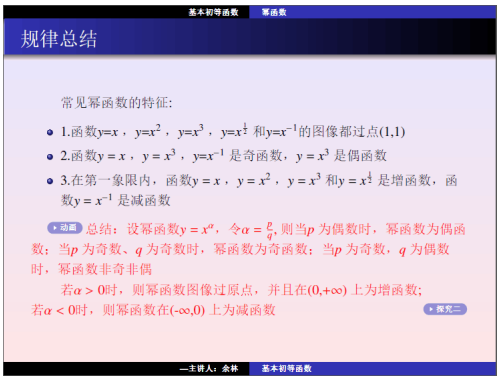

图 6: 第6张幻灯片

第六张幻灯片上的内容也是依次显示出来, 并且为幂函数特征的知识点总结, 对最主要结论给予了红色标注, 让学生重视该结论. 在此设置按钮为“动画”的超链接, 链接到第八张幻灯片上, 先让学生观察幂函数多个不同类型的图像, 在观察图像的基础上, 从而引导学生正确认识幂函数的奇偶性及其特征；同时还设置了按钮为”探究二“的超链接, 链接到第七张幻灯片上, 从而避免了第六张幻灯片的重复播放.

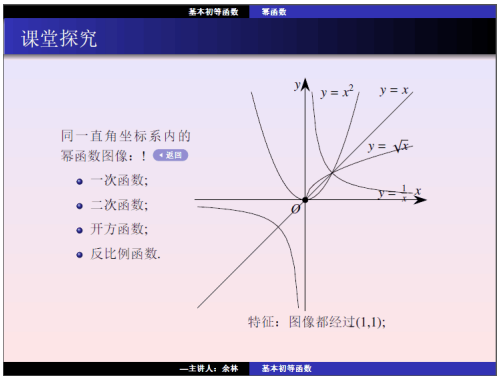

图 7: 第7张幻灯片

第七张幻灯片主要是对学生展示同一坐标系中几种常见幂函数的图像, 使得学生对幂函数的特征有比较直观全面的认识. 本张幻灯片主要作用为常见幂函数图像的播放, 使得在学生观察完图像后, 对填写第五张幻灯片上的表格有着提示与引导作用, 便于规律的总结. 本张幻灯片使用环境minipage 将其分为了左右两个部分, 幻灯片左边的函数名称可以使用列表使其与幻灯片上右边的相应图像同步播放出来, 最后在全部显示出来, 让学生有一个从单个图像到整体图像的全面认识. 此外, 本张幻灯片上设有超链接按钮“返回”, 将直接链接到第四张幻灯片上. 该张幻灯片的编写语言为：

```
\begin{frame}[label=tuxiang]
\frametitle{课堂探究}
```

```
\begin{minipage}{.3\textwidth}
同一直角坐标系内的幂函数图像：\breturn{fanhui}{返回}
\begin{itemize}
\item<1,5->一次函数;\\
\item<2,5->二次函数;\\
\item<3,5->开方函数;\\
\item<4->反比例函数.
\end{itemize}
\end{minipage}
\begin{minipage}{.59\textwidth}
\tikz[scale=.7]{
\uncover<1,5-> {\draw(-4,-4)--(4,4)node[left]{$y=x$};}
\uncover<2,5-> {\draw[domain=-2:2]plot (\x,{\x*\x})node[left]{$y=x^2$};}
\uncover<3,5-> {\draw[domain=0:4]plot(\x,{sqrt(\x)})node[left]{$y=\sqrt{x}$};}
\uncover<4-> {\draw[domain=1/4:4]plot(\x,1/\x)node[left]{$y=\frac{1}{x}$};
\draw[domain=-4:-1/4]plot(\x,1/\x);}
\axis[1]{-4.1,0}{4.2,0}{0,-4.1}{0,4.2};}\\
特征：图像都经过(1,1);
\end{minipage}
\end{frame}
```

其中命令\axis为自定义命令.

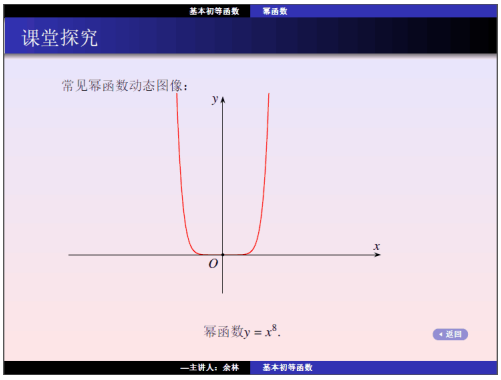

图 8: 第8张幻灯片

第八张幻灯片主要为学生直观呈现幂函数的动态图像, 加强学生对幂函数内在规律的理解与认识, 揭示幂函数一般性质和其他函数的外在联系, 此处主要用于幂函数奇偶性正确认知的引导与启发, 在学生观察完动态图像后, 进行相应的规律总结. 此处单击鼠标即可播放动态图像. 而点击“返回”按钮则会回到第六张幻灯片上, 在LaTeX编写的语言为:

```
常见幂函数动态图像：\\
\begin{animateinline}[
begin={\begin{tikzpicture}[scale=1,>=stealth]
%\clip (-2,-2) rectangle (2,2); %只在这个区域作图
},end={\axis[.3]{-4,0}{4,0}{0,-1}{0,4};
\end{tikzpicture}}]{1}
\multiframe{8}{Rpos=8+-1}{
\useasboundingbox(-4.2, -2.2)rectangle (4.2,4.2);
\draw[red,thick,samples=100,domain=-1.2:1.2] plot(\x,{(\x)^\Rpos});
\node at (.5,-2){幂函数$y=x^\Rpos$};}
\newframe
\multiframe{6}{Rpos=2+1}{
\useasboundingbox(-4.2,-2.2) rectangle (4.2,4.2);
\draw[blue,thick,samples=100,domain=0:1.2] plot(\x^\Rpos,\x);
\node at (.5,-2){幂函数$y=x^{1/\Rpos}$};}
\newframe
\multiframe{7}{Rpos=2+1}{
\useasboundingbox(-4.2, -2.2)rectangle (4.2,4.2);
\draw[cyan,thick,samples=100,domain=.8:4] plot({\x^(-\Rpos)},\x);
\node at (.5,-2){幂函数$y=x^{-1/\Rpos}$};}
\newframe
\multiframe{2}{Rpos=1+1}{
\useasboundingbox(-4.2,-2.2) rectangle (4.2,4.2);
\draw[black,thick,samples=100,domain=0.2:4] plot(\x,{\x^(-\Rpos)});
\node at (.5,-2){幂函数$y=x^{-\Rpos}$};}
\newframe
\multiframe{7}{Rpos=3+1}{
\useasboundingbox(-4.2,-2.2) rectangle (4.2,4.2);
\draw[black,thick,samples=100,domain=0.8:4] plot(\x,{\x^(-\Rpos)});
\node at (.5,-2){幂函数$y=x^{-\Rpos}$};}
\end{animateinline}
```

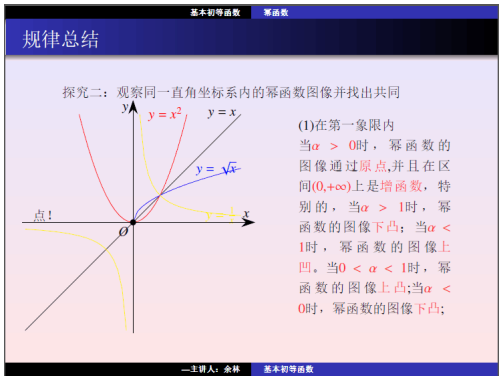

图 9: 第9张幻灯片

第九张幻灯片的设置与第六张类似, 是对幂函数特征的另一个探究, 研究其凹凸性及增长情况, 此处幂函数中不同范围的α取值, 其对应幂函数图像用了不同的颜色进行绘画, 便于学生观察规律, 从而让学生对幂函数有充分的认识. 本张幻灯片排版设计方式与第六张差不多.

第十张幻灯片设置与第九张相同, 其目的也与第七张相同, 是对幂函数特征的全面深入研究. 通过引导学生认识图像中对称情况来引出第二条规律, 最后再观察幂函数的幂指数与函数上升趋势的关系.

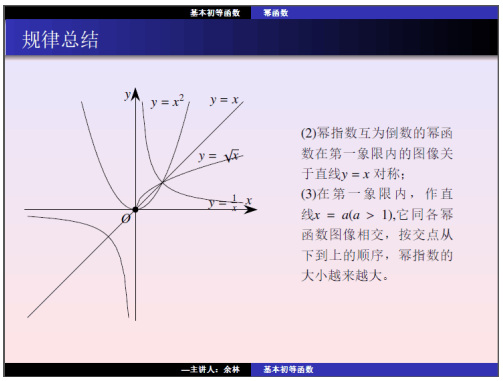

图 10: 第10张幻灯片

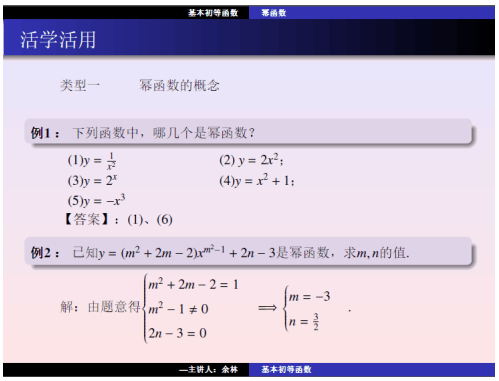

图 11: 第11张幻灯片

第十一张幻灯片是对幂函数第一类题型幂函数的定义的例题讲解, 幻灯片的内容会先播放题目, 再播放答案, 并且每个题目都会在前一道题目讲解完毕后才会出现. 在此处教师可以引导学生先进行探讨, 对题目进行分析, 最后再来公布答案. 本张幻灯片上的例题题目使用了定理环境进行着重, 便于学生简历相应模型和对例题解答的区分.

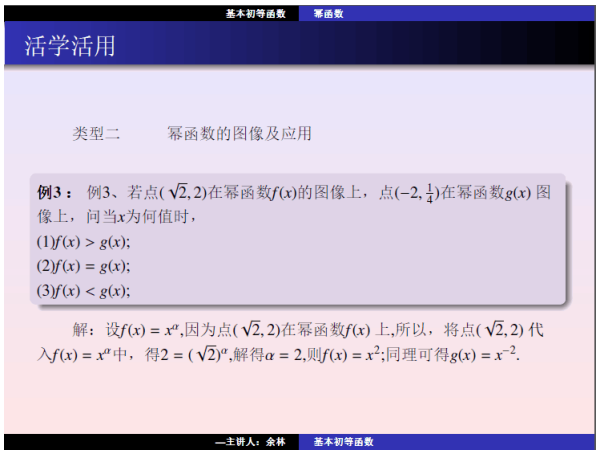

图 12: 第12张幻灯片

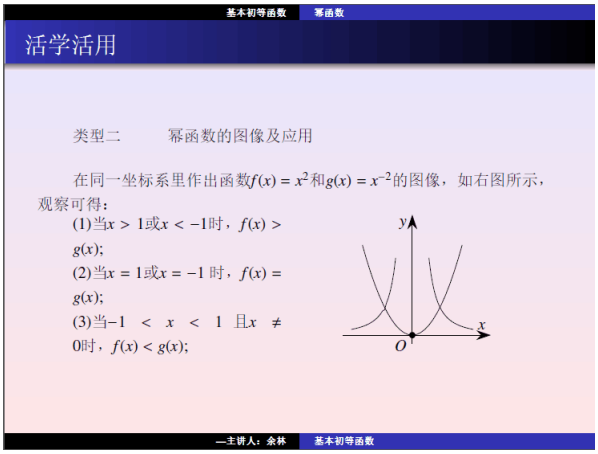

图13: 第13张幻灯片

第十二张与第十三张幻灯片是对幂函数第二类题型函数的图像及运用的例题讲解, 这两张的设置与前面第十一张的设置相同. 只是在第十三张的第一句话下使用了一个minipage环境, 左边输入文字, 右边绘画图形. 其设计意图与第九张幻灯片相同.

第十四张到第十五张幻灯片是对幂函数第二类题型的讲解, 利用幂函数的单调性对函数值大小的比较. 幻灯片上的动画为先播放题目, 再播放答案, 并且答案内容会依次播放. 其讲解方式与前面的例题相同. 但第十五张幻灯片中画出了例4中三个小题中对应的函数图像, 在学生进行思考时可给予图像帮助学生, 引导学生利用合理的方法进行解答, 然后再给出答案. 第十四张幻灯片上的第一个按钮“图像”会链接到第十五张幻灯片上, 使得学生可以先观察图像进行思考；而第二个按钮“例四”会链接到第十六张幻灯片上, 进行例5的讲解. 第十五张幻灯片上的“返回”按钮将会回到第十四张幻灯片上, 完成答案显示.

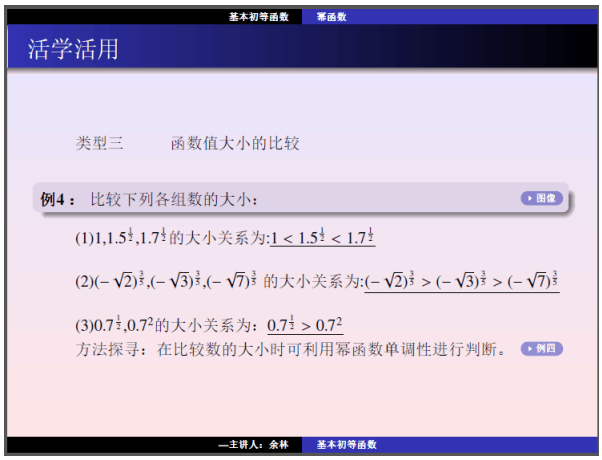

图 14: 第14张幻灯片

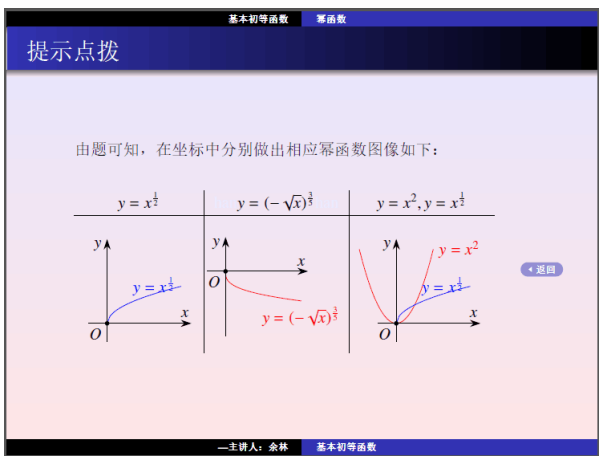

图15: 第15张幻灯片

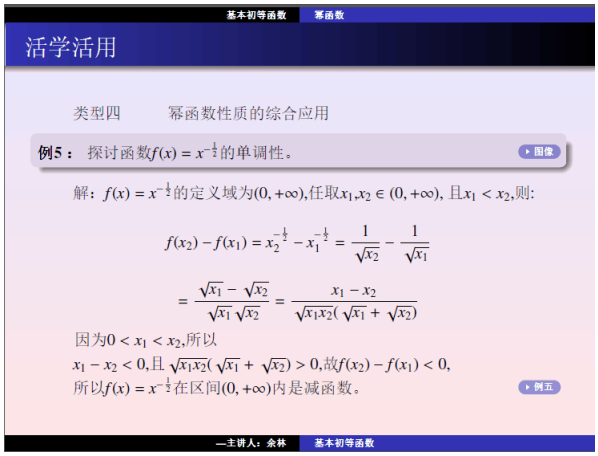

图 16: 第16张幻灯片

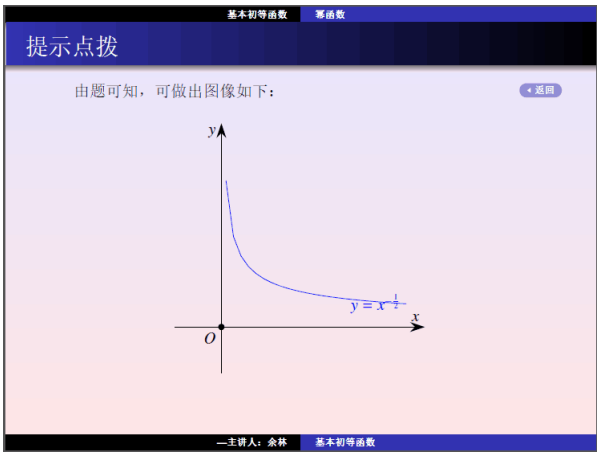

图17: 第17张幻灯片

第十六张到第十七张幻灯片是对幂函数第三类题型的讲解, 利用幂函数综合性质的运用. 幻灯片上的画面播放次序为先播放题目, 再播放答案, 并且答案内容会依次播放出来. 其讲解方式与前面的例题相同, 对学生加以引导疏通. 其中第十七张幻灯片中画出了例5中对应的函数图像, 在学生进行思考时可给予图像帮助学生, 先让学生认识到该题中的函数为减函数, 在该基础上用相应的证明方法完成证明. 第十六张幻灯片上的第一个按钮“图像”会链接到第十七张幻灯片上, 使得学生可以从结论出发, 寻找结论成立的证明方法；而第二个按钮“例五”会链接到第十八张幻灯片上, 进行例6的讲解. 第十七张幻灯片上的“返回”按钮将会回到第十六张幻灯片上, 完成相应的证明过程, 给出解答.

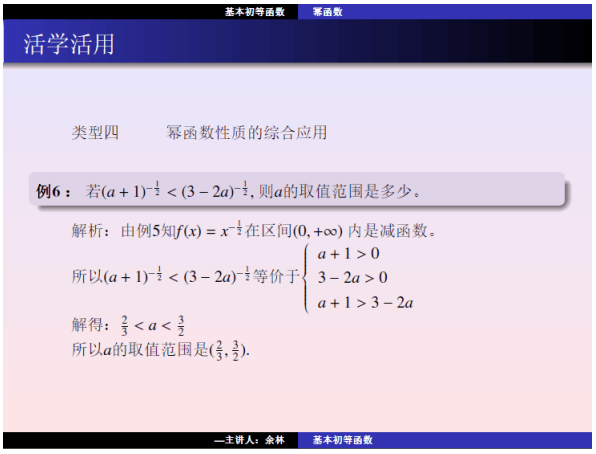

图 18: 第18张幻灯片

第十八张幻灯片是对幂函数第三类题型, 例5的补充讲解. 其幻灯片的播放形式与设计意图与前面的例题相同. 其中“例6”解析的生成言为：

1 解析：\uncover<2-> {由例5知$f(x)=x^{-\frac{1}{2}}$ 在区间$(0,+\infty)$内是减函数. }\\

```
\uncover<3-> {所以$(a+1)^{-\frac{1}{2}} <(3-2a)^ {-\frac{1}{2}}$等价于}
\uncover<4-> { $\left\{\begin{array}{l}
a+1>0\\
3-2a>0\\
a+1>3-2a
\end{array}\right.$} \\
\uncover<5-> {解得：$ \frac{2}{3} <a< \frac{3}{2}$\\ 所以$a$的取值范围是
$(\frac{2}{3},\frac{3}{2})$.}
```

通过以上的课件教学展示, 第一可以体现出课件的简洁优美, 容易吸引读者眼球；第二可以体现出使用LaTeX制作课件的便捷与功能的一体化, 可操作性较强；第三可以让教师在教学设计过程中不受诸多因素的约束, 随意发挥创新力, 从而提高教学能力与学生的学习效率.

在如今的信息时代, 课件制作方式多种多样, 但使用LaTeX制作课件不为是一件明智之举的事. 使用LaTeX制作课件, 不仅仅在操作上便捷与高效, 在呈现上美观简洁, 吸引读者, 而且还能丰富教师的设计意图, 达到创新的效果, 提高学生的学习效率与教师的教学质量. 由于LaTeX强大的功系统, 很多功能都可以使用命令和语言得以实现, 所以若想要设计出更为绚丽多彩, 有特色的课件, 则需要读者在LaTeX的使用上花费更多的心思与努力. 本文只是对使用LaTeX制作课件的一个简单介绍, 还存在诸多不住之处, 望读者能够见谅, 并在此基础上进行修正.

参考文献

[1] 陈志杰, 赵书钦, 李树钧等. LaTeX入门与提高[M]. 北京: 高等教育出版社, 2006.

[2] 邓建松译. LaTeX 2_ε用户手册中文版[M]. 北京: 高等教育出版社, 2006.

[3] 谷红霞. “指数函数及其性质”(第三课时)教学设计[J]. 中国数学教育, 2017(2): 16-18.

[4] 胡伟. LaTeX 2_ε完全学习手册[M]. 北京: 清华大学出版社, 2011.

[5] 乐小良. 高中三角教学中渗透数学思想方法的研究[D]. 湖南科技大学, 2018.

[6] 李平. LaTeX 2_ε及常用宏包使用指南[M]. 北京: 清华大学出版社, 2004.

致谢

我们的

附录

sdkfj

dfdf

dsf