

---

# 软件工程作业管理系统 概要设计

---

	人员	日期
拟制	张三 李四 王五	yyyy-mm-dd
评审人	•	yyyy-mm-dd
批准	•	yyyy-mm-dd
签发	•	yyyy-mm-dd

## 摘 要

本文是软件工程需求规格说明书模板，修改自于中国科学技术大学本硕博毕业论文  $\LaTeX$  模板示例文件，该模板由 `zepinglee` 和 `seisman` 创建，遵循中国科学技术大学的论文写作规范，适用于撰写学士、硕士和博士学位论文。

本文档最后一章演示如何使用  $\LaTeX$  的一些基本命令以及本模板提供的一些特殊功能，模板的选项及详细用法请参考模板说明文档 `ustcthesis.pdf`。请在提交之前把最后一掌实例注释掉。

**关键词：** 软件工程   中国科学技术大学   学位论文    $\LaTeX$  通用模板   学士  
硕士   博士   示例文档   模板说明文档

表 1   缩略词清单

缩略语	英文全名	中文解释
<code>c</code>	<code>d</code>	<code>e</code>

## 目 录

摘要 .....	
第 1 章 引言 .....	6
1.1 编写目的 .....	6
1.2 项目背景 .....	6
1.3 术语 .....	7
第 2 章 任务概述 .....	8
2.1 目标 .....	8
2.2 开发与运行环境 .....	8
2.2.1 开发环境的配置 .....	8
2.2.2 测试环境的配置 .....	8
2.2.3 运行环境的配置 .....	8
2.3 需求概述 .....	8
2.4 条件与限制 .....	10
第 3 章 总体设计 .....	11
3.1 软件描述 .....	11
3.1.1 前台 .....	11
3.1.2 后台 .....	11
3.2 处理流程 .....	14
3.2.1 总体流程 .....	14
3.2.2 系统基本流程 .....	14
3.2.3 R.FUNC.BB.001 用户登陆 .....	14
3.2.4 R.FUNC.BB.002 作业/实验查询 .....	15
3.2.5 R.FUNC.BB.003 作业/实验/通知发布 .....	15
3.2.6 R.FUNC.BB.004 作业/实验/通知修改删除 .....	15
3.2.7 R.FUNC.BB.005 作业/实验/通知提交 .....	15
3.2.8 R.FUNC.BB.006 作业/实验查看与批改 .....	15
3.2.9 R.FUNC.BB.007 成绩录入 .....	15
3.2.10 R.FUNC.BB.008 成绩查询 .....	16

3.2.11 R.FUNC.BB.009 成绩统计和调整 .....	16
3.2.12 R.FUNC.BB.0010 资源上传 .....	16
3.2.13 R.FUNC.BB.0012 资源下载与浏览 .....	16
3.2.14 R.FUNC.BB.0013 新建日程 .....	16
3.2.15 R.FUNC.BB.0014 日程显示与提醒 .....	16
3.2.16 R.FUNC.BB.0015 讨论区讨论 .....	17
3.2.17 R.FUNC.BB.0016 新建博文 .....	17
3.2.18 R.FUNC.BB.0018 学习笔记文本添加 .....	17
3.3 功能结构设计 .....	17
3.3.1 整体结构 .....	17
3.3.2 用户与服务器功能结构 .....	18
3.4 功能需求与程序代码的关系 .....	23
第 4 章 接口设计 .....	26
4.1 外部接口 .....	26
4.1.1 支付宝接口 .....	26
4.2 内部接口 .....	26
第 5 章 数据结构设计 .....	27
5.1 逻辑结构设计 .....	27
5.1.1 服务器端数据结构 .....	27
5.1.2 客户端数据结构 .....	32
5.1.3 用户端数据结构 .....	34
5.2 物理结构设计 .....	34
5.3 数据结构与程序模块的关系 .....	35
第 6 章 数据库设计 .....	36
6.1 数据库环境说明 .....	36
6.2 数据库的命名规则 .....	36
6.3 逻辑设计 .....	36
6.3.1 逻辑设计的 E-R 图 .....	36
6.4 物理设计 .....	36
6.4.1 数据库产品 .....	36
6.4.2 实体属性、类型、精度 .....	36
6.5 安全性设计 .....	39
6.6 数据库管理与维护说明 .....	40

第 7 章 界面设计·····	41
7.1 客户端界面 ·····	41
7.2 服务器端界面 ·····	41
7.3 登录界面 ·····	41
7.4 xxx 功能界面·····	41
第 8 章 出错处理设计·····	42
8.1 数据库出错处理 ·····	42
8.2 某模块失效处理 ·····	42
第 9 章 安全保密设计·····	43
第 10 章 维护设计 ·····	44
第 11 章 图片 ·····	45
11.1 示例 ·····	45
11.2 带图注的图 ·····	45
第 12 章 表格 ·····	47
12.1 A Simple Table·····	47
12.2 长表格 ·····	47
第 13 章 算法环境 ·····	49
第 14 章 代码环境 ·····	51
第 15 章 引用文献标注·····	52
15.1 著者-出版年制标注法·····	52
15.2 顺序编码制标注法 ·····	52
15.3 其他形式的标注 ·····	53
参考文献 ·····	54

## 图目录

11.1 测试图片 .....	45
11.2 带图注的图片 .....	46

## 表目录

1 缩略词清单 .....	
1.1 术语表 .....	7
2.1 开发环境的配置 .....	8
2.2 测试环境的配置 .....	9
2.3 运行环境的配置 .....	9
5.1 数据结构与程序代码的关系表 .....	35
6.1 日程表 .....	37
6.2 作业/实验发布表 .....	37
6.3 作业/实验提交表 .....	37
6.4 用户信息表 .....	38
6.5 成绩表 .....	38
6.6 笔记表 .....	38
6.7 讨论区表 .....	39
6.8 资源共享表 .....	39
12.1 这里是表的标题 .....	47
12.2 长表格演示 .....	47

## 第 1 章 引言

### 1.1 编写目的

在本项目的前一阶段，也就是需求分析阶段，已经将系统用户对本系统的需求做了详细的阐述，这些用户需求已经在上一阶段中对不同用户所提出的不同功能，实现的各种效果做了调研工作，并在需求规格说明书中得到详尽得叙述及阐明。

本阶段在系统的需求分析的基础上，对学习/教学辅助软件系统做概要设计。主要解决了实现该系统需求的程序模块设计问题。包括如何把该系统划分成若干个模块、决定各个模块之间的接口、模块之间传递的信息，以及数据结构、模块结构的设计等。在以下的概要设计报告中将对在本阶段中对系统所做的所有概要设计进行详细的说明，在设计过程中起到了提纲挈领的作用。

在下一阶段的详细设计中，程序设计员可参考此概要设计报告，在概要设计所做的模块结构设计的基础上，对系统进行详细设计。在以后的软件测试以及软件维护阶段也可参考此说明书，以便于了解在概要设计过程中所完成的各模块设计结构，或在修改时找出在本阶段设计的不足或错误。

### 1.2 项目背景

为了能更加高效的开展教学工作，充分利用各类信息化工具，市场上出现了各类教学辅助平台和学习辅助工具，因为其便捷和高效受到人们的青睐。在广泛的应用之中也的确方便了广大学生和教职人员。

但是仍然普遍存在以下这些问题：

- 各产品功能分散不完善，往往需要同时交错使用多个系统
- 功能落后，更新不及时
- 用户界面不美观，用户体验差
- 处理不够智能，系统内部模块协同差，使用不便捷



为此，我们提出了此项目，希望能够整合现有各系统的各项功能并在其基础之上加强模块协同，融合新的设计理念，加强用户体验。最终能够更好的服务于广大教师的教学工作并能够为同学的学习提供辅助，提高学习效率。

### 1.3 术语

[列出本文档中所用到的专门术语的定义和外文缩写的原词组]

表 1.1 术语表

缩写、术语	解释
c	d

## 第 2 章 任务概述

本系统的目标是实现一个 xxx 系统，包括客户端、服务器端两个部分。  
客户端面向 xxx 用户，为用户提供 xx 和 xx 服务。

### 2.1 目标

实现 xxx 系统，实现需求规格说明书中所描述的 xx 功能、xxx 功能和 xxx 功能，并且保证系统的健壮性和数据安全。

### 2.2 开发与运行环境

#### 2.2.1 开发环境的配置

表 2.1 开发环境的配置

类别	标准配置	最低配置
计算机硬件	基于 x86 结构的 CPU 主频 $\geq 2.4\text{GHz}$ 内存 $\geq 8\text{G}$ 硬盘 $\geq 200\text{G}$	基于 x86 结构的 CPU 主频 $\geq 1.6\text{GHz}$ 内存 $\geq 512\text{M}$ 硬盘 $\geq 2\text{G}$
计算机软件	Linux (kernel version $\geq 4.10$ ) GNU gcc (version $\geq 6.3.1$ )	Linux (kernel version $\geq 3.10$ ) GNU gcc (version $\geq 5.4$ )
网络通信	至少要有一块可用网卡 能运行 IP 协议栈即可	至少要有一块可用网卡 能运行 IP 协议栈即可
其他	采用 MySQL 数据库	采用 MySQL 数据库

#### 2.2.2 测试环境的配置

#### 2.2.3 运行环境的配置

### 2.3 需求概述

功能需求包括：

表 2.2 测试环境的配置

类别	标准配置	最低配置
计算机硬件	基于 x86 结构的 CPU 主频 $\geq 2.4\text{GHz}$ 内存 $\geq 8\text{G}$ 硬盘 $\geq 200\text{G}$	基于 x86 结构的 CPU 主频 $\geq 1.6\text{GHz}$ 内存 $\geq 512\text{M}$ 硬盘 $\geq 2\text{G}$
计算机软件	Linux (kernel version $\geq 4.10$ ) GNU gcc (version $\geq 6.3.1$ )	Linux (kernel version $\geq 3.10$ ) GNU gcc (version $\geq 5.4$ )
网络通信	至少要有一块可用网卡 能运行 IP 协议栈即可	至少要有一块可用网卡 能运行 IP 协议栈即可
其他	采用 MySQL 数据库	采用 MySQL 数据库

表 2.3 运行环境的配置

类别	标准配置	最低配置
计算机硬件	基于 x86 结构的 CPU 主频 $\geq 2.4\text{GHz}$ 内存 $\geq 8\text{G}$ 硬盘 $\geq 200\text{G}$	基于 x86 结构的 CPU 主频 $\geq 1.6\text{GHz}$ 内存 $\geq 512\text{M}$ 硬盘 $\geq 2\text{G}$
计算机软件	Linux (kernel version $\geq 4.10$ ) GNU gcc (version $\geq 6.3.1$ )	Linux (kernel version $\geq 3.10$ ) GNU gcc (version $\geq 5.4$ )
网络通信	至少要有一块可用网卡 能运行 IP 协议栈即可	至少要有一块可用网卡 能运行 IP 协议栈即可
其他	采用 MySQL 数据库	采用 MySQL 数据库

## 2.4 条件与限制

本节至少要与需求说明文档中相关章节相一致。

## 第 3 章 总体设计

### 3.1 软件描述

系统包括前台和后台两个部分。

#### 3.1.1 前台

前台主要功能是：

##### 3.1.1.1 学生端

学生用户需要是相应课程的注册用户，具体需要经过相应老师和教学管理人员的同意，成为相应资源的使用者和课程参与者，同时可参与讨论与资源分享。

##### 3.1.1.2 教师端

教师身份经过认证，教师可以注册生成相应的课程，使用软件辅助教学，进行资源分享和作业等的布置等等。

#### 3.1.2 后台

后台主要功能是：创建维护用户数据库，存储用户上传的课件作业等信息，对用户各种请求做出响应。

表 3.1 主要功能

编号	功能	简介
R.FUNC.BB.001	用户登陆	用户在 IOS、Android、Web Browse 输入账号和密码登录对应账户。登录成功后自动与服务器同步用户数据。

续下页

表 3.1 主要功能（续）

编号	功能	简介
R.FUNC.BB.002	作业/实验查询	作业/实验查询用于查询作业/实验情况。学生可以查到本人相关课程的作业情况。教师可以查询相应班级所有学生每次作业的提交情况。教学管理人员可以查询到自己所负责所有学生相应的作业提交情况。
R.FUNC.BB.003	作业/实验/通知发布	作业/实验/通知发布仅供教师端和管理人员使用。可以用于教师发布自己负责课程相关的作业、实验或通知。
R.FUNC.BB.004	作业/实验/通知修改删除	仅供教师和管理人员使用。用于修改或删除已发布的作业/实验/通知
R.FUNC.BB.005	作业/实验/通知提交	仅供学生使用。学生提交完成的作业、实验或需要提交材料的通知。
R.FUNC.BB.006	作业/实验查看与批改	仅供教师端使用。作业/实验查看与批改用与教师查看学生相应的作业或实验。
R.FUNC.BB.007	成绩录入	成绩输入仅供教师端使用。可以用于登记考试/作业等等的成绩。
R.FUNC.BB.008	成绩查询	成绩查询用于查询已经录入的成绩。学生可以查到本人相应课程的所有成绩。老师可以查到相应班级所有同学的成绩。教学管理人员可以查询自己所负责所有学生的相应成绩。

续下页

表 3.1 主要功能（续）

编号	功能	简介
R.FUNC.BB.009	成绩统计和调整	成绩统计功能教学管理人员与老师都拥有，成绩调整只有老师有相应的权限。成绩统计可用于老师和教学管理人员了解教学情况以及学生的学业状况。成绩调整可以供老师修订成绩同时也可以方便按照一定的规则对于成绩进行统一调整使得成绩更有一般性并能更好的体现学生的水平。
R.FUNC.BB.010	资源上传	资源上传可用于教师分享课件等教学辅助材料。同时同学自身也可以分享有用材料。空间分为共享和私有两部分。私有空间可以用作个人文件的托管和同步。
R.FUNC.BB.012	资源下载与浏览	资源下载用于获取远端分享的相关资源
R.FUNC.BB.013	新建日程	本功能用于生成新的日程安排，辅助用户合理安排各项任务
R.FUNC.BB.014	日程显示与提醒	日程提醒显示用于提醒自身的任务安排，合理安排时间。避免遗忘重要事项。
R.FUNC.BB.015	讨论区讨论	讨论区可以发布新的主题用于进行问题讨论，增强对知识的理解和应用，可在讨论区中发布新主题，修订，回复。
R.FUNC.BB.016	新建博文	博文主要用于用户分享学习经验等等认为有意义的东西，相比讨论区更加系统。

续下页

表 3.1 主要功能（续）

编号	功能	简介
R.FUNC.BB.017	课程考试排布	课程考试排布功能主要用于辅助教学管理人员合理安排相关活动，减少冲突节约时间。
R.FUNC.BB.018	学习笔记文本添加	学生用户在课堂或者课下记录笔记，选择创建新的笔记或者修改已经存在的笔记，进入文本编辑界面进行，编辑完成后点击保存到本地或者上传，点击返回键退出。
R.FUNC.BB.019	学习笔记课件修改	学生用户在老师上传的课件上使用文本框或者画图功能添加批注笔记。
R.FUNC.BB.020	学习笔记搜索	对笔记（文本模块和课件模块）进行检索，可根据创建时间，修改时间，课程，文本进行搜索。
R.FUNC.BB.021	学习笔记删除	删除某一笔记
R.FUNC.BB.022	学习笔记共享	用户与其他用户共享学习笔记。

## 3.2 处理流程

### 3.2.1 总体流程

此处应当有一个图和对应的描述。

### 3.2.2 系统基本流程

此处应当有一个图和对应的描述。

### 3.2.3 R.FUNC.BB.001 用户登陆

客户端发送查询请求到服务器用户数据库，匹配该用户邮箱或学号和密码，匹配成功则返回成功登录的信息，并同步数据，进入登录后页面。匹配失败则返回失败信息，并显示账户不存在或密码错误。若为新用户，客户端发送新建请求



到服务器数据库，检查是否为新用户，若可创建则返回允许创建信息，失败则返回无法创建新用户。

### 3.2.4 R.FUNC.BB.002 作业/实验查询

已登录用户在作业实验模块中提交查询请求，提交的表单中指明查找的关键字，由客户端和服务端从本地存储和服务器存储的数据文件进行搜索，对于符合关键字的搜索结果返回给用户，如果没有结果则显示无结果。

### 3.2.5 R.FUNC.BB.003 作业/实验/通知发布

教师在客户端作业实验端编辑要发布的信息的内容，并包括截止日期，附件等信息，上传到服务器，服务器判断用户发来的数据，将附件存入相应空间，将其他内容存入数据库中。再由服务器端向相应课程的学生客户端发出更新请求，更新日程和作业实验页面。

### 3.2.6 R.FUNC.BB.004 作业/实验/通知修改删除

教师通过客户端删除通知后，客户端删除本地缓存文件，同时在全局标签文件中删除信息，搜索日程中对应的作业实验并删除。编辑结束后上传删除信息，服务器进行相同的处理，并广播给相应课程的学生客户端，发出删除通知。

### 3.2.7 R.FUNC.BB.005 作业/实验/通知提交

学生将作业/实验内容从本地提交。服务器接受消息，更新数据库，并将收到的文件存储，向客户端发送结果信息。

### 3.2.8 R.FUNC.BB.006 作业/实验查看与批改

教师向服务器发出请求，下载班内同学作业到本地，或者在线浏览，批改，之后将批改结果上传到服务器端的相应位置，更新数据库中作业批改的信息，和相应课程同学的作业成绩。

### 3.2.9 R.FUNC.BB.007 成绩录入

教师通过客户端将学生成绩上传到服务器端的相应位置，更新数据库中作业或者考试信息，和相应课程同学的作业成绩，根据用户或者其他模块的输入构

造满足后台数据存储结构的信息并且存储。服务器向学生端发送数据，包括课程成绩，教师的批改情况等信息。

### 3.2.10 R.FUNC.BB.008 成绩查询

学生通过客户端发送请求，包括课程信息等，服务器端在成绩统计系统中查找学生的课程成绩或作业成绩，将信息发送给客户端，并更新数据。

### 3.2.11 R.FUNC.BB.009 成绩统计和调整

教师端通过客户端查看学生成绩，并进行调整，服务器解析用户的调整需求看是否合法，如果合法则对数据库进行更新。

### 3.2.12 R.FUNC.BB.0010 资源上传

首先检查上传数据是否有效合法。不合法或者失效给出警告。合法则计算MD5 查询数据库中是否已经存在相同资源，存在则只建立相应的链接，不存在则上传文件至远程数据库。

### 3.2.13 R.FUNC.BB.0012 资源下载与浏览

用户从服务器端获取数据，根据数据格式以及用户平台进行相应的转码处理最后交付客户端直接进行显示。

### 3.2.14 R.FUNC.BB.0013 新建日程

学生通过客户端编辑日程信息并提交，服务器端检查用户提供信息是否合法。日程的输入有多种格式，如果用户输入的不是标准的各项信息，而是纯文本等等，则首先尝试从中解析出事件地点描述等关键信息，尝试构建日程，成功则存入数据库，失败返回报错。成功后在日程管理系统中讲日程信息存入数据库，操作成功则返回成功信息到客户端。

### 3.2.15 R.FUNC.BB.0014 日程显示与提醒

客户端按设定频率检索当前日程数据库中的相关信息，比较当前日期，如果有满足提醒条件的则给用户发送提醒。

### 3.2.16 R.FUNC.BB.0015 讨论区讨论

用户在讨论区模块发送新建回复等请求，服务器端检查用户提供信息是否合法，然后将回复内容存入数据库中。客户端再从数据库中获取相关信息，处理渲染之后显示。

### 3.2.17 R.FUNC.BB.0016 新建博文

用户从讨论区模块创建新的博文，并在文本编辑器中编辑好文章，创建成功则将新的文章按用户选择存入本地存储，或存入服务器，并设定其他用户的浏览权限。

### 3.2.18 R.FUNC.BB.0018 学习笔记文本添加

用户从笔模块创建新的笔记，并在文本编辑器中编辑好文章，创建成功则将新的笔记按用户选择存入本地存储，或存入服务器，并设定其他用户的浏览权限。服务器检查用户提供信息是否合法。然后将其中的图片公式等等转换格式存入数据库中。

## 3.3 功能结构设计

### 3.3.1 整体结构

此处应当有一个图和对应的描述。系统如果像微内核那样，划分成核心模块和若干个子系统，此处应当有图示及说明，然后后续几个节应当描述这几个子系统。如果系统像宏内核，那应当说明有哪些紧密联系的模块，并在后续几个节内描述这些模块。

表 3.2 整体结构

模块编号	模块名称	子功能结构
M.MODULE.BB.001	日程管理模块	日程管理功能结构 日程查询功能结构 新建日程功能结构 日程显示功能结构
M.MODULE.BB.002	作业实验模块	作业实验功能结构

续下页

表 3.2 整体结构（续）

模块编号	模块名称	子功能结构
M.MODULE.BB.003	成绩管理模块	作业实验查询功能结构 新建作业功能结构 修改与删除作业功能结构 作业实验提交功能结构 成绩管理功能结构 成绩查询功能结构 成绩导入功能结构 成绩调整功能结构
M.MODULE.BB.004	讨论区模块	讨论区功能结构 发布主题功能结构 讨论区查询功能结构 修改与删除主题功能结构
M.MODULE.BB.005	资源共享模块	资源共享功能结构 资源查询功能结构 资源上传功能结构 资源下载功能结构
M.MODULE.BB.006	学习笔记模块	学习笔记功能结构 笔记上传功能结构 笔记查询功能结构 笔记修改与删除功能结构
M.MODULE.BB.007	用户管理模块	用户管理功能结构 登录请求功能结构
M.MODULE.BB.008	通知管理模块	通知管理功能结构

### 3.3.2 用户与服务器功能结构

客户端和服务端的功能结构具体描述。

### 3.3.2.1 MODULE.BB.001 日程管理功能结构

提供日程管理的功能部件，本功能结构实现对编辑按钮、查询按钮、导出按钮、删除选项、已完成选项的布局 and 接口实现。

子功能结构：

MODULE.BB.015 日程查询功能结构

MODULE.BB.018 新建日程功能结构

MODULE.BB.019 日程显示功能结构

### 3.3.2.2 MODULE.BB.002 作业实验功能结构

提供作业实验的功能部件，本功能结构实现对编辑按钮、查询按钮、下载按钮、导出按钮、删除选项、已完成选项的布局 and 接口实现。

子功能结构：

MODULE.BB.020 新建作业功能结构

MODULE.BB.021 修改与删除作业功能结构

MODULE.BB.022 作业实验提交功能结构

MODULE.BB.009 作业实验查询功能结构

### 3.3.2.3 MODULE.BB.003 成绩管理功能结构

提供成绩管理的功能部件，本功能结构实现对查询成绩按钮、上传成绩按钮、已完成选项的布局 and 接口实现。

子功能结构：

MODULE.BB.023 成绩导入功能结构

MODULE.BB.024 成绩调整功能结构

MODULE.BB.010 成绩查询功能结构

### 3.3.2.4 MODULE.BB.004 讨论区功能结构

提供讨论区的功能部件，本功能结构实现对新建主题按钮、编辑主题按钮、查询按钮、上传按钮、删除选项、已完成选项的布局 and 接口实现。

子功能结构：

MODULE.BB.025 修改与删除主题功能结构

MODULE.BB.016 发布主题功能结构

MODULE.BB.011 讨论区查询功能结构

### 3.3.2.5 MODULE.BB.005 资源共享功能结构

提供成绩的功能部件，本功能结构实现对编辑按钮、查询按钮、下载选项、上传选项、删除选项、已完成选项的布局 and 接口实现。

子功能结构：

MODULE.BB.026 资源上传功能结构

MODULE.BB.028 资源下载功能结构

MODULE.BB.012 资源查询功能结构

### 3.3.2.6 MODULE.BB.006 学习笔记功能结构

提供成绩的功能部件，本功能结构实现对编辑按钮、查询按钮、上传按钮、下载按钮、在线浏览、删除选项、已完成选项的布局 and 接口实现。

子功能结构：

MODULE.BB.017 笔记上传功能结构

MODULE.BB.013 笔记查询功能结构

MODULE.BB.027 笔记修改与删除功能结构

### 3.3.2.7 MODULE.BB.007 用户管理功能结构

服务器端提供用户信息管理的功能部件，本功能结构实现对用户信息的导入，查询，删除，编辑、已完成选项的布局 and 接口实现。

子功能结构：

MODULE.BB.014 登录请求功能结构

### 3.3.2.8 MODULE.BB.008 通知管理功能结构

根据本地数据库查询找到需要提醒的时间，实现一个 Android Fragment 用于时间的设置并触发一个 Thread 用于该提醒的计时。

### 3.3.2.9 MODULE.BB.009 作业实验查询功能结构

作业实验查询功能由触发查询按钮的回调函数实现，本功能结构完成对查询关键字的分词、分析和转化，并完成查询和分类展示的部分。

### 3.3.2.10 MODULE.BB.010 成绩查询管理功能结构

成绩查询功能由触发查询按钮的回调函数实现，本功能结构完成对查询关键字的分词、分析和转化，并完成查询和分类展示的部分。

### 3.3.2.11 MODULE.BB.011 讨论区查询功能结构

讨论区查询功能由触发查询按钮的回调函数实现，本功能结构完成对查询关键字的分词、分析和转化，并完成查询和分类展示的部分。

### 3.3.2.12 MODULE.BB.012 资源查询功能结构

资源查询功能由触发查询按钮的回调函数实现，本功能结构完成对查询关键字的分词、分析和转化，并完成查询和分类展示的部分。

### 3.3.2.13 MODULE.BB.013 笔记查询功能结构

笔记查询功能由触发查询按钮的回调函数实现，本功能结构完成对查询关键字的分词、分析和转化，并完成查询和分类展示的部分。

### 3.3.2.14 MODULE.BB.014 登录请求功能结构

登录界面由一个 Android Activity 实现，并用 socket 的方式连接服务器确认登录情况。

### 3.3.2.15 MODULE.BB.015 日程查询功能结构

日程查询功能由触发查询按钮的回调函数实现，本功能结构完成对查询关键字的分词、分析和转化，并完成查询和分类展示的部分。

### 3.3.2.16 MODULE.BB.016 发布主题功能结构

布局在 `note.xml` 布局文件中实现。完成编辑界面的布局。编辑界面作为一个新的 `Android Activity` 实现。编辑界面完成文本编辑功能，并为添加图片、添加画图、添加书签的实现准备好接口。实现基本的时间记录功能，编辑结束后，笔记内容以 `xml` 的格式保存在本地 APP 文件下，每次客户端打开时读取 `xml` 文件并解析。

### 3.3.2.17 MODULE.BB.017 笔记上传功能结构

MODULE.BB.006 学习笔记功能结构后用户选择是否调用该功能结构，实现笔记的上传功能。

### 3.3.2.18 MODULE.BB.018 新建日程功能结构

布局在 `note.xml` 布局文件中实现。完成编辑界面的布局。编辑界面作为一个新的 `Android Activity` 实现。编辑界面完成文本编辑功能，并为添加图片、添加画图、添加书签的实现准备好接口。实现基本的时间记录功能，编辑结束后，日程内容以 `xml` 的格式保存在本地 APP 文件下，每次日程打开时读取 `xml` 文件并解析。

### 3.3.2.19 MODULE.BB.019 日程显示功能结构

本功能结构由一个 `Android Fragment` 实现结果的分类展示。

### 3.3.2.20 MODULE.BB.020 新建作业功能结构

布局在 `note.xml` 布局文件中实现。完成编辑界面的布局。编辑界面作为一个新的 `Android Activity` 实现。编辑界面完成文本编辑功能，并为添加图片、添加画图、添加书签的实现准备好接口。实现基本的时间记录功能，编辑结束后，作业内容以 `xml` 的格式保存在本地 APP 文件下，每次作业实验模块打开时读取 `xml` 文件并解析。



#### 3.3.2.21 MODULE.BB.021 修改与删除作业功能结构

完成编辑和删除作业功能，编辑结束后，作业内容以 xml 的格式保存在本地 APP 文件下，每次作业实验模块打开时读取 xml 文件并解析。

#### 3.3.2.22 MODULE.BB.022 作业实验提交功能结构

从本地资源管理器中选择要提交的作业实验文件上传到数据库中。

#### 3.3.2.23 MODULE.BB.023 成绩导入功能结构

教师调用该功能结构，实现成绩的上传功能。

#### 3.3.2.24 MODULE.BB.024 成绩调整功能结构

界面在.xml 文件中实现，并可在线编辑文件。

#### 3.3.2.25 MODULE.BB.025 修改与删除主题功能结构

完成编辑和删除作业功能，编辑结束后，作业内容以 xml 的格式保存在本地 APP 文件下，每次讨论区模块打开时读取 xml 文件并解析。

#### 3.3.2.26 MODULE.BB.026 资源上传功能结构

用户选择调用该功能结构，实现课件等的上传功能。

#### 3.3.2.27 MODULE.BB.027 笔记修改与删除功能结构

完成编辑和删除作业功能，编辑结束后，笔记内容以 xml 的格式保存在本地 APP 文件下，每次笔记模块打开时读取 xml 文件并解析。

#### 3.3.2.28 MODULE.BB.028 资源下载功能结构

用户选择调用该功能结构，实现课件等的下载功能。

### 3.4 功能需求与程序代码的关系

[此处指的是不同的需求分配到哪些模块去实现。可按不同的端拆分此表]

表 3.3 功能需求与程序代码的关系

功能需求编号	功能需求	功能结构编号	功能结构
R.FUNC.BB.001	用户登陆	MODULE.BB.007	用户管理功能结构
		MODULE.BB.014	登录请求功能结构
R.FUNC.BB.002	作业/实验查询	MODULE.BB.002	作业实验功能结构
		MODULE.BB.001	日程管理功能结构
		MODULE.BB.009	作业实验查询功能结构
R.FUNC.BB.003	作业/实验/通知发布	MODULE.BB.002	作业实验功能结构
		MODULE.BB.020	新建作业功能结构
R.FUNC.BB.004	作业/实验/通知修改删除	MODULE.BB.021	修改与删除作业功能结构
R.FUNC.BB.005	作业/实验/通知提交	MODULE.BB.022	作业实验提交功能结构
R.FUNC.BB.006	作业/实验查看与批改	MODULE.BB.002	作业实验功能结构
		MODULE.BB.003	成绩管理功能结构
R.FUNC.BB.007	成绩录入	MODULE.BB.023	成绩导入功能结构
R.FUNC.BB.008	成绩查询	MODULE.BB.003	成绩管理功能结构
		MODULE.BB.010	成绩查询管理功能结构
		MODULE.BB.008	通知管理功能结构
R.FUNC.BB.009	成绩统计和调整	MODULE.BB.003	成绩管理功能结构
		MODULE.BB.010	成绩查询管理功能结构
		MODULE.BB.002	作业实验功能结构
		MODULE.BB.024	成绩调整功能结构
R.FUNC.BB.010	资源上传	MODULE.BB.005	资源共享功能结构
		MODULE.BB.026	资源上传功能结构
R.FUNC.BB.012	资源下载与浏览	MODULE.BB.005	资源共享功能结构
		MODULE.BB.012	资源查询功能结构
		MODULE.BB.026	资源下载功能结构
R.FUNC.BB.013	新建日程	MODULE.BB.018	新建日程功能结构
R.FUNC.BB.014	日程显示与提醒	MODULE.BB.019	日程显示功能结构
		MODULE.BB.008	通知管理功能结构

续下页

表 3.3 整体结构（续）

功能需求编号	功能需求	功能结构编号	功能结构
R.FUNC.BB.015	讨论区讨论	MODULE.BB.004	讨论区功能结构
		MODULE.BB.025	修改与删除主题功能结构
R.FUNC.BB.016	新建博文	MODULE.BB.004	讨论区功能结构
		MODULE.BB.016	发布主题功能结构
R.FUNC.BB.017	课程考试排布	MODULE.BB.001	日程管理功能结构
		MODULE.BB.015	日程查询功能结构
R.FUNC.BB.018	学习笔记文本添加	MODULE.BB.006	学习笔记功能结构
R.FUNC.BB.019	学习笔记课件修改	MODULE.BB.027	笔记修改与删除功能结构
R.FUNC.BB.020	学习笔记搜索	MODULE.BB.013	学习笔记查询功能结构
		MODULE.BB.006	学习笔记功能结构
R.FUNC.BB.021	学习笔记删除	MODULE.BB.027	笔记修改与删除功能结构
		MODULE.BB.013	学习笔记查询功能结构
R.FUNC.BB.022	学习笔记共享	MODULE.BB.006	学习笔记功能结构
		MODULE.BB.013	学习笔记查询功能结构
		MODULE.BB.017	笔记上传功能结构

## 第 4 章 接口设计

### 4.1 外部接口

#### 4.1.1 支付宝接口

详细讲述不同的接口（查询状态、支付交易、获取回执等）

### 4.2 内部接口

内部模块/系统之间的交互的接口。

## 第 5 章 数据结构设计

### 5.1 逻辑结构设计

#### 5.1.1 服务器端数据结构

##### 5.1.1.1 客户端请求

请求以 `socket` 的形式以字节的格式传输到服务器端口。

代码 5.1 客户端请求

```
1  struct command{
2      Byte b;
3  }
```

##### 5.1.1.2 登录请求

解析结果是登录请求，请求字段分别为用户 `id` 和密码，两个都加密后进行传输。

代码 5.2 客户端请求

```
1  struct LoginCommand{
2      commandID id;
3      String userid;
4      String password;
5  }
```

##### 5.1.1.3 作业/实验/通知查询请求

解析结果为作业实验查询请求，请求字段为用户 `id` 和请求查询的作业/实验名称。

代码 5.3 作业/实验/通知查询请求

```
1  struct WorkQueryCommand{
2      commandID id;
3      String userid;
4      String workid;
5  }
```

#### 5.1.1.4 作业/实验/通知发布请求

向服务器发布作业/实验/通知。解析为发布请求，请求字段为用户名，之后为字节流，内容是作业/实验/通知的详细信息，具体解析工作交由作业/实验/通知功能结构完成。

代码 5.4 作业/实验/通知发布请求

```
1  struct WorkPublishCommand{
2      commandID id;
3      String userid;
4      Byte s;
5  }
```

#### 5.1.1.5 作业/实验/通知修改请求

修改已经发布的作业/实验/通知。解析为修改请求，请求字段为用户名以及需要修改的工作名，之后为字节流，内容为需要进行修改的详细内容，具体解析工作交由作业/实验/通知功能结构完成。

代码 5.5 作业/实验/通知修改请求

```
1  struct WorkAdjustCommand{
2      commandID id;
3      String userid;
4      String workid;
5      Byte s;
6  }
```

#### 5.1.1.6 作业/实验提交请求

提交所完成的作业/实验。解析为提交请求，请求字段为用户名以及待提交的作业，之后为字节流，内容为需要提交的具体成果，具体解析工作交由作业/实验/通知功能结构完成。

代码 5.6 作业/实验提交请求

```
1  struct WorkCommitCommand{
2      commandID id;
3      String userid;
4      String workid;
5      Byte s;
6  }
```

#### 5.1.1.7 作业/实验查看请求

查看所完成的作业/实验。解析为提交请求，请求字段为用户名以及待查看的作业。

代码 5.7 作业/实验查看请求

```

1      struct WorkCheckCommand{
2          commandID id;
3          String userid;
4          String workid;
5      }
    
```

#### 5.1.1.8 成绩录入请求

解析结果是录入成绩请求，请求字段是用户 id，作业 id，之后是字节流，内容为具体学生和成绩的信息，具体解析交由成绩录入功能模块处理。

代码 5.8 成绩录入请求

```

1      struct ScoreCommitCommand{
2          commandID id;
3          String userid;
4          String workid;
5          Byte s;
6      }
    
```

#### 5.1.1.9 成绩查询请求

解析结果是录入成绩请求，请求字段是用户 id，待查询用户 id，作业 id。

代码 5.9 成绩查询请求

```

1      struct ScoreCommitCommand{
2          commandID id;
3          String userid;
4          String queryid;
5          String workid;
6      }
    
```

#### 5.1.1.10 成绩统计调整请求

解析结果是成绩统计调整请求，请求字段是用户 id，作业 id，后面跟有字节流，内容为具体的更改或者统计信息。

代码 5.10 成绩统计调整请求

```

1      struct ScoreAdjustCommand{
2          commandID id;
3          String userid;
4          String workid;
5          Byte s;
6      }
    
```

#### 5.1.1.11 资源上传请求

解析结果是资源上传请求，请求字段是用户 id，后面跟有字节流，内容为上传的目的地址，上传文件的相关信息和具体资源数据。具体解析交由资源上传模块解析。

代码 5.11 资源上传请求

```

1      struct ResoureUploadCommand{
2          commandID id;
3          String userid;
4          Byte s;
5      }
    
```

#### 5.1.1.12 资源下载请求

解析结果是资源下载请求，请求字段是用户 id 和资源 id。

代码 5.12 资源下载请求

```

1      struct ResourceDownloadCommand{
2          commandID id;
3          String userid;
4          String resourceid;
5      }
    
```

#### 5.1.1.13 日程同步请求

解析结果是日程同步请求，请求字段是用户 id，后面跟有字节流，内容是本地等待同步的日程信息。具体解析交由日程同步模块处理。

代码 5.13 日程同步请求

```

1      struct ScheduleSynCommand{
2          commandID id;
3          String userid;
4          Byte s;
5      }
    
```



#### 5.1.1.14 讨论区新建/回复请求

解析结果是讨论区相关请求，请求字段是用户 id，后面跟有字节流，内容为具体的主题/回复信息。具体解析交由讨论区模块处理。

代码 5.14 讨论区新建/回复请求

```
1  struct NewDisCommand{
2      commandID id;
3      String userid;
4      Byte s;
5  }
```

#### 5.1.1.15 讨论区查询请求

解析结果是讨论区查询请求，请求字段是用户 id 和需要查询的主题。

代码 5.15 讨论区查询请求

```
1  struct DisQueryCommand{
2      commandID id;
3      String userid;
4      String thesis
5  }
```

#### 5.1.1.16 新建博文请求

解析结果是新建博文请求，请求字段是用户 id，后面跟有字节流，内容为具体博文内容。具体解析交由博文模块处理。

代码 5.16 新建博文请求

```
1  struct NewBlogCommand{
2      commandID id;
3      String userid;
4      Byte s;
5  }
```

#### 5.1.1.17 学习笔记同步请求

解析结果是学习笔记同步请求，请求字段是用户 id，后面跟有字节流，内容为本地待提交的笔记内容，具体交由学习笔记同步模块解析。

代码 5.17 学习笔记同步请求

```
1  struct NoteSynCommand{
```

```

2      commandID id;
3      String userid;
4      Byte s;
5      }

```

## 5.1.2 客户端数据结构

### 5.1.2.1 作业/实验信息

代码 5.18 作业/实验信息

```

1      struct WorkInfo{
2          String caption;
3          String content;
4          Data deadline;
5          String[] userid;
6          String publisher;
7      }

```

作业实验的信息包括标题、详细内容、提交截止时间、需要提交作业的用户以及发布者。

### 5.1.2.2 通知信息

代码 5.19 通知信息

```

1      struct NoticeInfo{
2          String caption;
3          String content;
4          String[] userid;
5          String publisher;
6      }

```

通知信息包括标题，详细内容，需要接受消息的用户以及消息的发布者。

### 5.1.2.3 个人成绩

代码 5.20 个人成绩

```

1      struct ScoreEntry{
2          int score;
3          String userid
4      }

```

每条成绩记录包括用户和其相应的成绩。

#### 5.1.2.4 成绩表

代码 5.21 成绩表

```
1  struct ScoreList{
2      String Test;
3      ScoreEntry[] score;
4  }
```

成绩记录中包括考试/作业名称和各个用户的成绩。

#### 5.1.2.5 资源信息

代码 5.22 资源信息

```
1  struct ResourceInfo{
2      String name;
3      int type;
4      int size;
5      Byte content;
6  }
```

资源信息包括资源名称，资源的类型，资源大小，资源内容等等。

#### 5.1.2.6 日程信息

代码 5.23 日程信息

```
1  struct ScheduleInfo{
2      Data data;
3      int duration;
4      String name;
5      String place;
6      String description;
7  }
```

每条日程包括日程的时间地点，事件的持续时间，事件的名称和详细描述。

#### 5.1.2.7 讨论区主题

代码 5.24 讨论区主题

```
1  struct DisThesis{
2      String title;
3      String content;
4      Data creationtime;
5      Vector<ReplyInfo> reply;
6      String author;
```

```
7 | }
```

每个讨论区主题包含创建时间、标题、内容、作者以及回复信息。

#### 5.1.2.8 讨论区回复

代码 5.25 讨论区回复

```
1 struct ReplyInfo{
2     String content;
3     String author;
4     Data time;
5 }
```

每条回复包括回复内容、回复者和时间信息。

#### 5.1.2.9 博文信息

代码 5.26 博文信息

```
1 struct BlogInfo{
2     String title;
3     String author;
4     Data time;
5     int type;
6 }
```

每条博文记录包括具体内容、作者、时间以及是否公开。

#### 5.1.2.10 学习笔记

代码 5.27 学习笔记

```
1 struct NoteData{
2     String name;
3     Data time;
4     String author;
5     String content;
6 }
```

每个学习笔记包括标题、内容、创建时间和作者信息。

### 5.1.3 用户端数据结构

## 5.2 物理结构设计

各个数据结构无特殊物理结构要求。

## 5.3 数据结构与程序模块的关系

[此处指的是不同的数据结构分配到哪些模块去实现。可按不同的端拆分此表]

表 5.1 数据结构与程序代码的关系表

·	模块 1	模块 2	模块 3
结构 1	·	Y	·
结构 2	·	Y	·
结构 3	·	Y	·
结构 4	Y	·	·
结构 5	·	·	Y

注：各项数据结构的实现与各个程序模块的分配关系

## 第 6 章 数据库设计

### 6.1 数据库环境说明

本系统的数据系统采用 MySQL 数据库系统。

在服务器端采用 Java 编程语言，使用 JDBC API。

### 6.2 数据库的命名规则

数据库的名字为 BBDatabase。

数据表的命名规则为 XX\_table，其中 XX 为数据表存储内容的英文单词，单词间用 \_ 连接，首字母大写，如果英文单词长度大于 5 则使用缩写。

如 HomeWork 缩写为 HW，Information 缩写为 Info。

表中每一个字段以该字段代表含义的小写英文单词命名，缩写同上。字段不加前缀。

### 6.3 逻辑设计

#### 6.3.1 逻辑设计的 E-R 图

图。

### 6.4 物理设计

#### 6.4.1 数据库产品

使用 Mysql 数据库。

#### 6.4.2 实体属性、类型、精度

##### 6.4.2.1 日程表：Sche\_table

主键：{UserID, ScheID}

表 6.1 日程表

列名	数据类型	可否为空	说明
UserID	Char	Not NULL	用户学号
ScheID	Int	Not NULL	日程编号
ScheInfo	VarChar	Not NULL	日程内容
DDL	Datetime	Not NULL	截止日期
Done	Bit	Not NULL	是否完成

表 6.2 作业/实验发布表

列名	数据类型	可否为空	说明
CourseID	Char	Not NULL	课程编号
HWID	Int	Not NULL	作业编号
HWInfo	VarChar	Not NULL	作业内容
DDL	Datetime	Not NULL	截止日期
AnnexAddr	VarChar	NULL	附件地址

#### 6.4.2.2 作业/实验发布表：HW\_Pub\_table

主键：{CourseID, HWID}

#### 6.4.2.3 作业/实验提交表：HW\_Sub\_table

表 6.3 作业/实验提交表

列名	数据类型	可否为空	说明
UserID	Char	Not NULL	学号
CourseID	Char	Not NULL	课程编号
HWID	Int	Not NULL	作业编号
HWInfo	VarChar	Not NULL	提交内容
AnnexAddr	VarChar	NULL	附件地址

主键：{UserID, CourseID, HWID}

#### 6.4.2.4 用户信息表：User\_Info\_table

表 6.4 用户信息表

列名	数据类型	可否为空	说明
UserID	Char	Not NULL	工号
Position	Char	Not NULL	职位
CourseID	Char	NULL	课程号

主键：{UserID}

#### 6.4.2.5 成绩表：Score\_table

表 6.5 成绩表

列名	数据类型	可否为空	说明
UserID	Char	Not NULL	学号
CourseID	Char	Not NULL	课程号
HWID	Int	Not NULL	作业号
Score	Int	Not NULL	成绩

主键：{UserID, CourseID, HWID}

#### 6.4.2.6 笔记表：Note\_table

表 6.6 笔记表

列名	数据类型	可否为空	说明
UserID	Char	Not NULL	学号
NoteID	Int	Not NULL	笔记号
NoteName	VarCahr	Not NULL	笔记名
Note	VarChar	Not NULL	笔记内容

主键：{UserID, NoteID}



#### 6.4.2.7 讨论区表：Note\_table

表 6.7 讨论区表

列名	数据类型	可否为空	说明
UserID	Char	Not NULL	学号
UserName	Char	Not NULL	姓名
ThemeName	VarCahr	Not NULL	主题名
ThemeAddr	VarChar	Not NULL	主题地址
ThemeTime	Datetime	Not NULL	发布时间

主键：{UserID, ThemeName}

#### 6.4.2.8 资源共享表：Share\_table

表 6.8 资源共享表

列名	数据类型	可否为空	说明
UserID	Char	Not NULL	工号
UserAuth	Char	Not NULL	权限
Name	VarCahr	Not NULL	资源名
Addr	VarChar	Not NULL	资源地址
ResourceID	Int	Not NULL	资源号

主键：{UserID, ResourceID}

### 6.5 安全性设计

管理默认用户: 在生产环境中，必须严格管理 sys 和 system 用户，必须修改其默认密码，禁止用该用户建立数据库应用对象。删除或锁定数据库测试用户。

数据库级用户权限设计: 必须按照应用需求，设计不同的用户访问权限。包括应用系统管理用户，普通用户等，按照业务需求建立不同的应用角色。用户访问另外的用户对象时，应该通过创建同义词对象 synonym 进行访问。

角色与权限: 确定每个角色对数据库表的操作权限, 如创建、检索、更新、删除等。每个角色拥有刚好能够完成任务的权限, 不多也不少。在应用时再为用户分配角色, 则每个用户的权限等于他所兼角色的权限之和。

应用级用户设计: 应用级的用户帐号密码不能与数据库相同, 防止用户直接操作数据库。用户只能用帐号登陆到应用软件, 通过应用软件访问数据库, 而没有其它途径操作数据库。用户密码管理: 用户帐号的密码必须进行加密处理, 确保在任何地方的查询都不会出现密码的明文。

## 6.6 数据库管理与维护说明

对于数据库的维护, 随时对数据库中的信息加以调试和保存备份。同样需要个工作人员进行系统的分析和用户的反馈, 对系统进行升级以及功能的完善。同时保证系统安全有序的运行。

## 第 7 章 界面设计

### 7.1 客户端界面

此处应当有一个简略的图，重点是展示你与用户交互的逻辑。（processon 上画一个不花时间）

### 7.2 服务器端界面

此处应当有一个简略的图。

### 7.3 登录界面

此处应当有一个简略的图。

### 7.4 xxx 功能界面

此处应当有一个简略的图。

## 第 8 章 出错处理设计

### 8.1 数据库出错处理

多重备份时，应采取何种策略，先利用哪一份备份；系统是否暂停服务等。

### 8.2 某模块失效处理

是否整个系统暂停服务，还是维持最小服务状态、如何尽快恢复服务还是删库跑路等。

## 第 9 章 安全保密设计

可能的内容包括保密性、是否采取加密传输、密钥如何分发和管理等。

## 第 10 章 维护设计

可能的内容包括数据库的日常备份、压缩、维护等。

## 第 11 章 图片

本章展示图片相关用法。

### 11.1 示例



图 11.1 测试图片

### 11.2 带图注的图



图 11.2 带图注的图片

注：the solid lines represent the time histogram of the spontaneous activities of an old monkey cell(gray) and a young monkey cell (black). The bin-width is 1



## 第 12 章 表格

### 12.1 A Simple Table

表 12.1 这里是表的标题

a	b
c	d

注：这里是表的注释

### 12.2 长表格

表 12.2 长表格演示

名称	说明	备注
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC

续下页

[illegible]

## 第 13 章 算法环境

模板中使用 `algorithm2e` 宏包实现算法环境。关于该宏包的具体用法，请阅读宏包的官方文档。

```

Data: this text

Result: how to write algorithm with LATEX2ε

1 initialization;
2 while not at end of this document do
3   read current;
4   if understand then
5     go to next section;
6     current section becomes this one;
7   else
8     go back to the beginning of current section;
9   end
10 end

```

算法 13.1: 算法示例 1

```

input : A bitmap  $Im$  of size  $w \times l$ 
output: A partition of the bitmap

1 special treatment of the first line;
2 for  $i \leftarrow 2$  to  $l$  do
3   special treatment of the first element of line  $i$ ;
4   for  $j \leftarrow 2$  to  $w$  do
5      $left \leftarrow \text{FindCompress}(Im[i, j - 1]);$ 
6      $up \leftarrow \text{FindCompress}(Im[i - 1, j]);$ 
7      $this \leftarrow \text{FindCompress}(Im[i, j]);$ 
8     if left compatible with this then //  $O(left, this) == 1$ 
9       if  $left < this$  then  $\text{Union}(left, this);$ 
10      else  $\text{Union}(this, left);$ 
11    end
12    if up compatible with this then //  $O(up, this) == 1$ 
13      if  $up < this$  then  $\text{Union}(up, this);$ 
14      // this is put under up to keep tree as
15      // flat as possible
16      else  $\text{Union}(this, up);$ 
17      // this linked to up
18    end
19  end
20 foreach element  $e$  of the line  $i$  do  $\text{FindCompress}(p);$ 
21 end

```

算法 13.2: 算法示例 2

## 第 14 章 代码环境

模板中使用 listings 宏包实现代码环境。详细用法见宏包的官方说明文档。

以下是代码示例，可以在文中任意位置引用??。

代码 14.1 示例代码

```
1 #include <stdio.h>
2
3 int main( )
4 {
5     printf("hello, \world\n");
6     return 0;
7 }
```

## 第 15 章 引用文献标注

### 15.1 著者-出版年制标注法

<code>\citetstyle{ustcauthoryear}</code>	
<code>\cite{knuth86a}</code>	⇒ Knuth (1986)
<code>\citet{knuth86a}</code>	⇒ Knuth (1986)
<code>\citet[chap.~2]{knuth86a}</code>	⇒ Knuth (1986, chap. 2)
<code>\citep{knuth86a}</code>	⇒ (Knuth, 1986)
<code>\citep[chap.~2]{knuth86a}</code>	⇒ (Knuth, 1986, chap. 2)
<code>\citep[see][ ]{knuth86a}</code>	⇒ (see Knuth, 1986)
<code>\citep[see][chap.~2]{knuth86a}</code>	⇒ (see Knuth, 1986, chap. 2)
<code>\citet*{knuth86a}</code>	⇒ Knuth (1986)
<code>\citep*{knuth86a}</code>	⇒ (Knuth, 1986)
<code>\citet{knuth86a,tlc2}</code>	⇒ Knuth (1986); Mittelbach et al. (2004)
<code>\citep{knuth86a,tlc2}</code>	⇒ (Knuth, 1986; Mittelbach et al., 2004)
<code>\cite{knuth86a, knuth84}</code>	⇒ Knuth (1984, 1986)
<code>\citet{knuth86a, knuth84}</code>	⇒ Knuth (1984, 1986)
<code>\citep{knuth86a, knuth84}</code>	⇒ (Knuth, 1984, 1986)

### 15.2 顺序编码制标注法

`\citetstyle{ustcnumerical}`

<code>\cite{knuth86a}</code>	$\Rightarrow$	[2]
<code>\citet{knuth86a}</code>	$\Rightarrow$	Knuth <sup>[2]</sup>
<code>\citet[chap.~2]{knuth86a}</code>	$\Rightarrow$	Knuth <sup>[2]</sup> , chap. 2 <sup>]</sup>
<code>\citep{knuth86a}</code>	$\Rightarrow$	[2]
<code>\citep[chap.~2]{knuth86a}</code>	$\Rightarrow$	[2] chap. 2
<code>\citep[see][]{knuth86a}</code>	$\Rightarrow$	see <sup>[2]</sup>
<code>\citep[see][chap.~2]{knuth86a}</code>	$\Rightarrow$	see <sup>[2]</sup> chap. 2
<code>\citet*{knuth86a}</code>	$\Rightarrow$	Knuth <sup>[2]</sup>
<code>\citep*{knuth86a}</code>	$\Rightarrow$	[2]
<code>\citet{knuth86a,tlc2}</code>	$\Rightarrow$	Knuth <sup>[2]</sup> , Mittelbach et al. <sup>[3]</sup>
<code>\citep{knuth86a,tlc2}</code>	$\Rightarrow$	[2,3]
<code>\cite{knuth86a, knuth84}</code>	$\Rightarrow$	[1,2]
<code>\citet{knuth86a, knuth84}</code>	$\Rightarrow$	Knuth <sup>[1, 2]</sup>
<code>\citep{knuth86a, knuth84}</code>	$\Rightarrow$	[1,2]
<code>\cite{knuth86a, knuth84, tlc2}</code>	$\Rightarrow$	[1–3]

### 15.3 其他形式的标注

<code>\citealt{tlc2}</code>	$\Rightarrow$	Mittelbach et al. <sup>3</sup>
<code>\citealt*{tlc2}</code>	$\Rightarrow$	Mittelbach, Goossens, Braams, and Carlisle <sup>3</sup>
<code>\citealp{tlc2}</code>	$\Rightarrow$	<sup>3</sup>
<code>\citealp*{tlc2}</code>	$\Rightarrow$	<sup>3</sup>
<code>\citealp{tlc2, knuth86a}</code>	$\Rightarrow$	<sup>2,3</sup>
<code>\citealp[pg.~32]{tlc2}</code>	$\Rightarrow$	<sup>3</sup> pg. 32
<code>\citenum{tlc2}</code>	$\Rightarrow$	3
<code>\citetext{priv.\ comm.}</code>	$\Rightarrow$	[priv. comm.]
<code>\citeauthor{tlc2}</code>	$\Rightarrow$	Mittelbach et al.
<code>\citeauthor*{tlc2}</code>	$\Rightarrow$	Mittelbach, Goossens, Braams, and Carlisle
<code>\citeyear{tlc2}</code>	$\Rightarrow$	2004
<code>\citeyearpar{tlc2}</code>	$\Rightarrow$	2004

## 参考文献

- Knuth D E. May 1984. Literate programming[J]. *The Computer Journal*. 27(2):97–111.
- Knuth D E. 1986. Computers and Typesetting: A The  $\text{\TeX}$ book[M]. Reading, MA, USA: Addison-Wesley.
- Mittelbach F, Goossens M, Braams J, et al. 2004. The  $\text{\LaTeX}$  Companion[M]. 2nd ed. Reading, MA, USA: Addison-Wesley.