

第一部分：数据结构课程设计题目

实验目的：

1. 数据结构课程设计是综合运用数据结构课程中学到的几种典型数据结构，以及程序设计语言（C 语言），自行实现一个较为完整的应用系统的设计与开发
2. 通过课程设计，自己通过系统分析、系统设计、编程调试，写实验报告等环节，进一步掌握应用系统设计的方法和步骤，灵活运用并深刻理解典型数据结构在软件开发中的应用。
3. 学会将知识应用于实际的方法，提高分析和解决问题的能力，增加综合能力。

学生信息管理系统：

- (1) 熟练掌握链表存储结构及其建立过程和常用操作；
- (3) 学会自己调试程序的方法并掌握一定的技巧

设计题目 1：文章编辑

[设计要求]：

- (1) 静态存储一页文章，每行最多不超过 80 个字符，共 N 行；
- (2) 分别统计出其中英文字母数和空格数及整篇文章总字数；
- (3) 统计某一字符串在文章中出现的次数，并输出该次数；
- (4) 删除某一子串，并将后面的字符前移；
- (5) 存储结构使用线性表，分别用几个子函数实现相应的功能。

[输出形式]：

- (1) 分行输出用户输入的各行字符；
- (2) 分 4 行输出"全部字母数"、"数字个数"、"空格个数"、"文章总字数"
- (3) 输出删除某一字符串后的文章；

设计题目 2：学生信息管理系统

[设计要求]：

- (1) 建立学生信息数据，包括学号、姓名、性别、三科成绩、出生时间、年龄。
- (2) 用链表存放学生信息；
- (3) 实现简单的菜单调用；
- (4) 程序的功能包括：

学生信息管理

*增加学生

*显示全部学生

*计算平均成绩

*删除一个学生

*按年级输出学生信息

*按性别输出学生信息

*按姓名检索学生

*按某一学科的成绩对所有学生依次排序

*结束程序运行

设计题目 3：图书室借阅管理系统

[问题描述]：

对图书馆的图书借阅业务管理作一个简单的模拟。该系统存储以下信息：书名、书目编号、作者名、库存册数(不包括已借出的) 和已借出册数；对于每一本书还要存储一个登陆号(这是区别每一本书的唯一标志)、当前借阅该书的读者姓名与借书证号以及借阅日期。

[功能要求]：

设计图书借阅信息管理的模拟程序。(分管理员和普通用户两类人群设计系统)

- (1) 采用顺序表或结构体链表存储结构。
- (2) 实现图书的创建、分类、插入、删除、修改、借阅等管理功能。
- (3) 实现图书的各种借阅查询和统计功能。
- (4) 考虑采用优化的查询和排序算法。
- (5) 其它完善性功能。

设计题目 4：学生选修课程系统

[问题描述]：

假定有 n 门课程，每门课程有课程编号，课程名称，课程性质，总学时，授课学时，实验或上机学时，学分，开课学期等信息，学生可按要求(如总学分不得少于 60) 自由选课。

[设计要求]：

试设计一选修课程系统，使之能提供以下功能：

- (1) 系统以菜单方式工作
- (2) 课程信息录入功能(课程信息用文件保存)
- (3) 课程信息浏览功能
- (4) 查询功能：
 - *按学分查询
 - *按课程性质查询

参考界面如下：

```
*****
                        学生选修课程管理系统
*****
*课程信息录入
*课程信息浏览
*课程查询
  1 学分
  2 课程性质
*程序结束
```

设计题目 5：学生成绩管理系统

[问题描述]

设计一个实用的小型学生成绩管理程序，具有查询，检索和删除功能，并且能够打开或修改指定文件及将多个文件组成一个文件。它完全使用类来实现，充分体现面向对象的程序设计特点。

[设计要求]：

主要功能：建立文件，文件的存取和显示，删除记录，查询记录。可以随时检索，删除，或增加记录，也可以保存或取消新的记录。

[功能要求]：

(1) 信息维护：

要求：学生信息数据要以文件的形式保存，能实现学生信息数据的维护。此模块包括子模块有：增加学生信息、删除学生信息、修改学生信息

(2) 信息查询：

要求：查询时可实现按姓名查询、按学号查询

(3) 成绩统计：

要求：A 输入任意的一个课程名(如数学) 和一个分数段(如 60--70) ，统计出在此分数段的学生情况。

设计题目 6：高校人事管理系统

[问题描述]：

某高校主要人员有：在职人员(行政人员、教师)、退休人员、返聘人员和临时工。现在，需要存储这些人员的人事档案信息：编号、姓名、性别、年龄、职务、职称、政治面貌、最高学历、任职时间、来院时间。

[设计要求]：

(1) 添加删除功能：能根据学院人事的变动情况，添加删除记录；

(2) 查询功能：能根据编号和姓名进行查询；

(3) 编辑功能(高级)：根据查询对相应的记录进行修改，并存储；

(4) 统计功能：能根据多种参数进行人员的统计(在职人数、党员人数、女工人数、高学历高级职称人数)；

(5) 保存功能：能对输入的数据进行相应的存储，要求重载插入和提取符以完成数据的保存和打开。

(6) 人员编号在生成人员信息时同时生成，每输入一个人员信息编号顺序加 1。

设计题目 7：学生考勤管理系统

[问题描述]：

考勤信息记录了学生的缺课情况，它包括：缺课日期、第几节课(连续多节课用 begin-end 的形式表示)、课程名称(课程名称中不会出现空格)、学生姓名、缺课类型(迟到、早退、请假及旷课)。

[设计要求]：

试设计考勤管理系统，使之能提供以下功能：

(1) 录入学生的缺课记录：从键盘输入数据(提示：为避免重复从键盘输入数据，测试时可先将数据存储在文件中，利用输入重定向功能读入)，输入格式为：

缺课日期 第几节课 课程名称 学生姓名 缺课类型

每行一条记录，例如：

2008-04-29 3-4 C++程序设计实验 张三 迟到

2008-04-28 3-4 C++程序设计 李四 旷课

(2) 修改某个学生的缺课记录：可以对缺课纪录的任意部分进行修改，然后显示一下修改后的纪录。

(3) 查询某个学生的缺课情况：查询结果按照日期升序排序，同一天内按照所缺课程的时间升序排序。

(4) 统计某段时间内(以天为单位)，旷课学生姓名及旷课节数，查询结果先按旷课节数降序排序，旷课节数相同的学生按姓名升序排序；

(5) 统计某段时间内，有学生旷课的课程及旷课人次，按旷课人次由多到少排序，旷课人次相同的课程按课程名称升序排序；

(6) 系统以菜单方式工作。(所谓菜单指用户可以自由选择所要执行的功能)

设计题目 8：通信录管理系统

设计出模拟手机通信录管理系统，实现对手机中的通信录进行管理。

[设计要求]：

(1) 查看功能：选择此功能时，列出下列三类选择。

A 办公类 B 个人类 C 商务类，当选中某类时，显示出此类所有数据中的姓名和电话号码)

(2) 增加功能：能录入新数据(一个结点包括：姓名、电话号码、分类(可选项有：A 办公类 B 个人类 C 商务类)、电子邮件)。

例如：杨春 13589664454 商务类 chuny@126.com

当录入了重复的姓名和电话号码时，则提示数据录入重复并取消录入；当通信录中 15 条信息时，存储空间已满，不能再录入新数据；录入的新数据能按递增的顺序自动进行条目编号。

(3) 拨号功能：能显示出通信录中所有人的姓名，当选中某个姓名时，屏幕上模拟打字机的效果依次显示出此人的电话号码中的各个数字，并伴随相应的拨号声音。

(4) 修改功能：选中某个人的姓名时，可对此人的相应数据进行修改

(5) 删除功能：选中某个人的姓名时，可对此人的相应数据进行删除，并自动调整后续条目的编号。

设计题目 9：银行定期储蓄管理系统

[问题描述]：

该系统存储以下信息：

(1) 每笔存款的储户信息和办理该笔存款的营业员信息，这些信息存放于存款文件中。其中，储户信息包括：帐号、姓名、密码、地址、储种(定期 1 年、3 年、5 年)，本金、收储日期、是否已挂失和挂失日期，营业员信息包括接待该储户的营业员的姓名与工号。

(2) 每笔取款的储户信息和办理该笔取款的营业员信息，这些信息存放于取款文件中。

[设计要求]：

系统功能要求如下：

(1) 创建存款文件(第一次输入储户信息时)。

(2) 创建取款文件(第一次办理取款时)。

(3) 接收储蓄。接受储户与营业员信息并将以上信息添加到存款文件中。

(4) 处理挂失。根据储户提供的帐号、姓名、密码、地址、储种，核查有无此项存款，有则对此帐号加入挂失标记，若无此项存款，则需判断存款是、否被领走还是未发生过这笔存款。

[其他要求]：

办理取款：

(1) 根据储户提供的存款单(上有帐号、姓名、储种、本金、日期) 判断是否到期，检查有否挂失，根据储户提供的密码判断是否正确。若以上判断通过执行后面 3 步。

(2) 在取款文件中添加这笔取款的储户和营业员的所有信息和取款日期，以便复查。

(3) 打印利息单，其内容包括帐号、姓名、储种、起息日期、支取日期、本金、利息和支取金额。利息计算方法如下：1 年到期利率 1.98%，2 年到期利率 2.25%，3 年到期利率 2.52%，5 年到期利率 2.79%，每逾期一天，每天利率 0.002%，若未到期提前支取，按每天利率 0.002%计，每月按 30 天计算，利息税按 20%扣除。

(4) 在存款文件删除这笔存款。

办理存款流程自行分析。

设计题目 10：约瑟夫生者死者游戏

[问题描述]：

约瑟夫生者死者游戏的大意是：30 个旅客同乘一条船，因为严重超载，加上风高浪大，危险万分；因此船长告诉乘客，只有将全船一半的旅客投入海中，其余人才能幸免遇难。无奈，大家只得同意这种办法，并议定 30 个人围成一圈，由第一个人开始，依次报数，数到第 9 人，便把他投入大海中，然后从他的下一个人数起，数到第 9 人，再将他投入大海，如此循环，直到剩下 15 个乘客为止。问哪些位置是将被扔下大海的位置。

[设计要求]：

本游戏的数学建模如下：假设 n 个旅客排成一个环形，依次顺序编号 1, 2, ..., n 。从某个指定的第 1 号开始，沿环计数，每数到第 m 个人就让其出列，且从下一个人开始重新计数，继续进行下去。这个过程一直进行到剩下 k 个旅客为止。

本游戏的要求用户输入的内容包括：

- (1) 旅客的个数，也就是 n 的值；
- (2) 离开旅客的间隔数，也就是 m 的值；
- (3) 所有旅客的序号作为一组数据要求存放在某种数据结构中。

本游戏要求输出的内容是包括

- (1) 离开旅客的序号；
- (2) 剩余旅客的序号；

所以，根据上面的模型分析及输入输出参数分析，可以定义一种数据结构后进行算法实现。

设计题目 11：约瑟夫双向生死游戏

[问题描述]：

约瑟夫双向生死游戏是在约瑟夫生者死者游戏的基础上，正向计数后反向计数，然后再正向计数。

具体描述如下：30 个旅客同乘一条船，因为严重超载，加上风高浪大，危险万分；因此船长告诉乘客，只有将全船一半的旅客投入海中，其余人才能幸免遇难。无奈，大家只得同意这种办法，并议定 30 个人围成一圈，由第一个人开始，顺时针依次报数，数到第 9 人，便把他投入大海中，然后从他的下一个人数起，逆时针数到第 5 人，将他投入大海，然后从他逆时针的下一个人数起，顺时针数到第 9 人，再将他投入大海，如此循环，直到剩下 15 个乘客为止。问哪些位置是将被扔下大海的位置。

[设计思路]：

本游戏的数学建模如下：假设 n 个旅客排成一个环形，依次顺序编号 1, 2, ..., n 。从某个指定的第 1 号开始，沿环计数，数到第 m 个人就让其出列，然后从第 $m+1$ 个人反向计数到 $m-k+1$ 个人，让其出列，然后从 $m-k$ 个人开始重新正向沿环计数，再数 m 个人后让其出列，然后再反向数 k 个人后让其出列。这个过程一直进行到剩下 q 个旅客为止。

本游戏的要求用户输入的内容包括：

- (1) 旅客的个数，也就是 n 的值；
- (2) 正向离开旅客的间隔数，也就是 m 的值；
- (3) 反向离开旅客的间隔数，也就是 k 的值；
- (4) 所有旅客的序号作为一组数据要求存放在某种数据结构中。

设计题目 12：停车场管理系统

[问题描述]：

设停车场是一个可以停放 n 辆汽车的南北方向的狭长通道，且只有一个大门可供汽车进出。汽车在停车场内按车辆到达时间的先后顺序，依次由北向南排列(大门在最南端，最先到达的第一辆车停放在车场的最北端)，若车场内已停满 n 辆车，那么后来的车只能在门外的便道上等候，一旦有车开走，则排在便道上的第一辆车即可开入；当停车场内某辆车要离开时，在它之后进入的车辆必须先退出车场为它让路，待该辆车开出大门外，其它车辆再按原次序进入车场，每辆停放在车场的车在它离开停车场时必须按它停留的时间长短交纳费用。试为停车场编制按上述要求进行管理的模拟程序。要求程序输出每辆车到达后的停车位置(停车场或便道上)，以及某辆车离开停车场时应缴纳的费用和它在停车场内停留的时间。

[设计思路]：

停车场的管理流程如下：

(1) 当车辆要进入停车场时，检查停车场是否已满，如果未满，则车辆进入停车场；如果停车场已满，则车辆进入便道等候。

(2) 当车辆要求出栈时，先让在它之后进入停车场的车辆退出停车场为它让路，再让该车退出停车场，让路的所有车辆再按其原来进入停车场的次序进入停车场。之后，再检查在便道上是否有车等候，有车则让最先等待的那辆车进入停车场。

设计题目 13：魔王语言解释

[问题描述]：

有一个魔王总是使用自己的一种非常精练而抽象的语言讲话，没有人能听得懂，但他的语言是可以逐步解释成人能听懂的语言，因为他的语言是由以下两种形式的规则由人的语言逐步抽象上去的：

$$(1) \alpha \rightarrow \beta_1 \beta_2 \cdots \beta_m$$

$$(2) (\theta \delta_1 \delta_2 \cdots \delta_n) \rightarrow \theta \delta_n \theta \delta_{n-1} \cdots \theta \delta_1 \theta$$

在这两种形式中，从左到右均表示解释。试写一个魔王语言的解释系统，把他的话解释成人能听得懂的话。

[设计要求]：

用下述两条具体规则和上述规则形式(2)实现。设大写字母表示魔王语言的词汇；小写字母表示人的语言词汇；希腊字母表示可以用大写字母或小写字母代换的变量。魔王语言可含人的词汇。

$$(1) B \rightarrow tAdA$$

$$(2) A \rightarrow sae$$

[测试数据]：

B(chngxz)B 解释成 tsaedsacezegexenehetsaedsac

若将小写字母与汉字建立下表所示的对应关系,则魔王说的话是“天上一只鹅地上一只鹅鹅追鹅赶鹅下鹅蛋恨鹅天上一只鹅地上一只鹅”。

t	d	s	a	e	z	g	x	n	h
天	地	上	一只	鹅	追	赶	下	蛋	恨

[实现提示]:

将魔王的语言自右至左进栈,总是处理栈顶字符。若是开括号,则逐一出栈,将字母顺序入队列,直至闭括号出栈,并按规则要求逐一出队列再处理后入栈。应首先实现栈和队列的基本操作。

设计题目 14: 航空客运订票系统

[问题描述]:

航空客运订票的业务活动包括:查询航线、客票预订和办理退票等。试设计一个航空客运订票系统,以使上述业务可以借助计算机来完成。

[设计要求]:

(1) 每条航线所涉及的信息有:终点站名、航班号、飞机号、飞行周日(星期几)、乘员定额、余票量、已订票的客户名单(包括姓名、订票量、舱位等级 1, 2 或 3)以及等候替补的客户名单(包括姓名、所需票量);

(2) 系统能实现的操作和功能如下:

①查询航线:根据旅客提出的终点站名输出下列信息:航班号、飞机号、星期几飞行,最近一天航班的日期和余票额;

②承办订票业务:根据客户提出的要求(航班号、订票数额)查询该航班票额情况,若尚有余票,则为客户办理订票手续,输出座位号;若已满员或余票额少于订票额,则需重新询问客户要求。若需要,可登记排队候补;

③承办退票业务:根据客户提供的情况(日期、航班),为客户办理退票手续,然后查询该航班是否有人排队候补,首先询问排在第一的客户,若所退票额能满足他的要求,则为他办理订票手续,否则依次询问其他排队候补的客户。

[实现提示]:

两个客户名单可分别由线性表和队列实现。为查找方便,已订票客户的线性表应按客户姓名有序,并且,为插入和删除方便,应以链表作存储结构。由于预约人数无法预计,队列也应以链表作存储结构。整个系统需汇总各条航线的情况登录在一张线性表上,由于航线基本不变,可采用顺序存储结构,并按航班有序或按终点站名有序。每条航线是这张表上的一个记录,包含上述 8 个域、其中乘员名单域为指向乘员名单链表的头指针,等候替补的客户名单域为分别指向队头和队尾的指针。

设计题目 15: 电文的编码和译码

[问题描述]:

从键盘接收一串电文字符,输出对应的 huffman 编码。同时能翻译由 huffman 编码生成的代码串,输出对应的电文字符串

[设计要求]:

(1) 构造一棵 huffman 树

(2) 实现 huffman 编码,并用 huffman 编码生成的代码串进行译码

(3) 程序中字符和权值是可变的,实现程序的灵活性

[数据结构]

本课程设计使用结构体数组或链表作为数据结构来存储哈夫曼编码树及其编码

设计题目 16：家族关系查询系统

[问题描述]:

建立家族关系数据库，实现对家族成员关系的相关查询

[设计要求]:

- (1) 建立家族关系并能存储到文件中
- (2) 实现家族成员的添加
- (3) 可以查询家族成员关系的双亲、祖先、兄弟、孩子和后代等信息

[数据结构]

本课程设计使用的数据结构有树状结构和队列。树状结构采用三叉链表实现，队列采用链式队列或顺序队列实现

设计题目 17：地铁建设问题

[问题描述]:

某城市要在其各个辖区之间修建地铁来加快经济发展，但是由于建设地铁的费用昂贵，因此需要合理安排地铁的建设路线，使乘客可以沿地铁到达各个辖区，并使总的建设费用最小。

[设计要求]:

- (1) 从包含各辖区的地图文件中读入辖区名称和各辖区间的直接距离
- (2) 根据读入的各辖区的距离信息，计算出应该建设那些辖区间的地铁路线
- (3) 输出应该建设的地铁路线及所需建设的总里程信息

[数据结构]

本课程设计使用的数据结构是无向图。无向图采用邻接矩阵或邻接表作为存储结构

设计题目 18：安排教学计划

[问题描述]:

学校每个学期开设的课程是有先后顺序的，如计算机专业：开设数据结构课程之前，必须先开设 C 语言程序设计和离散数学课程，这种课程开设的先后顺序关系称为先行、后继课程关系，现在需要根据计划根据给定的课程信息及课程之间的先行、后继关系，合理安排出开设各门课程的先后顺序。

[设计要求]:

- (1) 对输入的课程先行、后继关系如果存在回路关系时应提示出现错误
- (2) 根据读入的课程信息及其先行、后继关系，计算出安排教学计划的序列
- (3) 输出教学计划和安排顺序或给出错误提示信息

[数据结构]

本课程设计使用的数据结构是有向图和栈。可采用邻接表或者邻接矩阵作为有向图的存储结构来实现，栈选用顺序栈或者链栈均可。

设计题目 19：校园导航系统

[问题描述]:

当我们参观某校园时，就会遇到这样一个问题：从当前所处位置出发去校园另外某个位置，要走什么样的路线距离最近？本课程设计实例在给出校园各主要建筑的名称信息及有路线连通的建筑物之间的距离的基础上，利用校园导航系统计算出给定的起点到终点之间的距离最近的行进路线

[设计要求]:

- (1) 从地图文件中读取校园主要建筑信息及建筑间的距离信息
- (2) 计算出给定的起点到终点之间的距离最近的行进路线
- (3) 输出该路线（包含路过哪些建筑）及其总距离
- (4) 若输入错误，则给出提示信息

[数据结构]

本课程设计使用的数据结构是有向网络。采用邻接表或邻接矩阵作为有向网络的存储结构。

设计题目 20：魔方阵

[问题描述]:

魔方阵是一个古老的智力问题,它要求在一个 $m \times m$ 的矩阵中填入 $1 \sim m^2$ 的数字(m 为奇数),使得每一行、每一列、每条对角线的累加和都相等,如下图所示:

15	8	1	24	17
16	14	7	5	23
22	20	13	6	4
3	21	19	12	10
9	2	25	18	11

6	1	8
7	5	3
2	9	4

[设计要求]

- (1) 输入魔方阵的行数 m , 要求 m 为奇数, 程序对所输入的 m 作简单的判断, 如 m 有错, 能给出适当的提示信息
- (2) 实现魔方阵
- (3) 输出魔方阵

[数据结构]

本课程设计使用的数据结构是数组。

设计题目 21：电话号码查询系统

[问题描述]:

设计散列表实现电话号码查找系统。

[设计要求]:

- (1) 设每个记录有下列数据项: 电话号码、用户名、地址;
- (2) 从键盘输入各记录, 分别以电话号码和用户名为关键字建立散列表;
- (3) 在散列函数确定的前提下, 尝试不同类型处理冲突的方法, 考察平均查找长度的变化;
- (4) 查找并显示给定电话号码的记录;
- (5) 查找并显示给定用户名的记录。
- (6) 设计不同的散列函数, 比较冲突率

[数据结构]

本课程设计采用散列表来实现。

设计题目 22：走迷宫游戏

[问题描述]:

程序开始运行时显示一个迷宫地图，迷宫中央有一只老鼠，迷宫的右下方有一个粮仓。游戏的任务是使用键盘上的方向键操纵老鼠在规定的时间内走到粮仓处。

[设计要求]:

- 1) 老鼠形象可辨认，可用键盘操纵老鼠上下左右移动；
- 2) 迷宫的墙足够结实，老鼠不能穿墙而过；
- 3) 正确检测结果，若老鼠在规定时间内走到粮仓处，提示成功，否则提示失败；
- 4) 添加编辑迷宫功能，可修改当前迷宫，修改内容：墙变路、路变墙；
- 5) 找出走出迷宫的所有路径，以及最短路径。

设计题目 23：身份证信息管理系统

[问题描述]:

建立一个身份证信息管理系统，能够进行身份证信息的录入，查找，**要求考虑查找效率**

[设计要求]:

- 1) 能够进行身份证号码及相关信息的录入，相关信息包括姓名、地址、手机号；
- 2) 能够快速进行身份证号码的查询，并输入相关信息；
- 3) 可以修改身份证号码对应的其他信息，如姓名、地址；
- 4) 可以完成身份证信息的删除；

[数据结构]

本课程设计用二叉排序树实现。

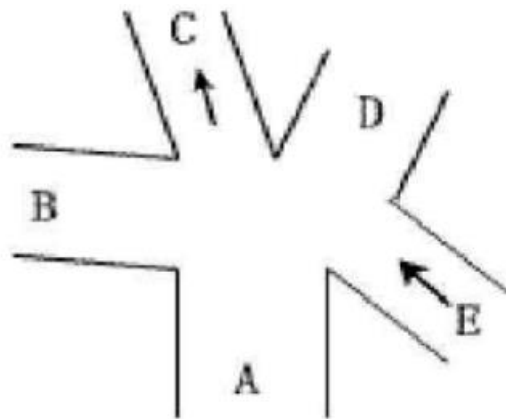
设计题目 24：五叉路口交通灯的管理系统

1. 设计目的:

- (1) 构造解决实际问题的图的模型
- (2) 熟悉邻接矩阵和邻接表这两种存在结构的特点及适用范围，能够根据应用问题的特点和要求，选择合适的存储结构
- (3) 掌握图顶点的着色问题及其解决方法

2. 问题描述:

设计一个交通信号灯的管理系统：一个五叉路口如下图所示，在这个路口共有 5 条道路相交，其中 C 和 E 是单行线，其他为双行线。在该路口需要设置几种颜色的交通灯才能既使车辆相互之间不碰撞，又能达到车辆的最大流通？



设计题目 25：大整数运算

1. 问题描述：

本题要求完成一个支持长度为 100 位以下的十进制的大整数运算系统，能够完成大整数的加法、减法、乘法三种功能。

大整数通常指超出计算机系统整数类型所能表达的范围的整数，比如几十、几百甚至成千上万位的大整数。

2. 设计要求：

(1) 由于大整数的位数比较多，从键盘输入比较麻烦，且容易出错，因此大整数运算系统应该具备从文本文件中读入大整数的功能

(2) 大整数的四则运算的结果也是大整数，若在屏幕上输出不便于用户查看，因此大整数运算系统应具备显示结果的前 m 位，显示结果的后 m 位功能；

(3) 能够对两个大整数进行加、减、乘 3 种运算

(4) 用户可以选择不同的“基”来表示大整数，任意一个 10 进制的整数都可以表示为

$$\begin{aligned} N &= (a_{n-1}a_{n-2} \dots a_i \dots a_1a_0)_{10} \\ &= a_{n-1} * 10^{n-1} + a_{n-2} * 10^{n-2} + \dots + a_i * 10^i + \dots + a_1 * 10^1 + a_0 * 10^0 \\ &= \sum_{i=0}^{n-1} a_i * 10^i \end{aligned}$$