

数据结构课程设计任务书

[课程设计基本要求]

1. 原则上可以 1 - 3 位同学组成实验小组，进行分工合作，但必需保证每位组员都充分参与实验过程，每位组员应对实验程序的结构、算法、主要技术完全掌握，方可参加实验验收。但一小组内领队成员外的其他参与成员无法获得中等以上评级。
2. 要求从下面的课程设计任务书中，任选一道题目进行设计和开发。
3. 课程设计时间从第 9 周开始至 16 周，要求在考试之前全部验收结束。原则上，课程设计只验收一次。若实验没有达到规定的要求，不可再次申请验收，故请大家务必确认程序正确（程序代码和运行结果）后，再申请验收。

[实验报告内容]

实验报告必须包括以下内容：

1. 实验报告封面。封面必须给出题目、班级、姓名、学号、和完成日期，如果是多人完成的，必须写明所有人员的班级、姓名和学号，并写明各自的主要工作，标明课题负责人。
2. 实验题目和要求。可以直接复制任务书的内容。
3. 设计思路：包括系统总体设计、系统功能设计、类的设计和主程序的设计。
 - (1) 系统总体设计：要求对系统的技术思路和数据结构进行概要性说明和解释；
 - (2) 系统功能设计：要求说明系统的所有功能，建议采用框图或表格等方式来表示，并辅以文字说明；
 - (3) 类的设计：包括所有类的设计、各个类之间的关系和主要成员函数的设计等。建议采用合适的方式(如，框图或表格)来表示类的设计和主要成员函数的功能；要求以图示方式来表示各个类之间的关系；对于关键和主要的成员函数，要求以程序流程图来表达其实现过程，同时可以适当粘贴关键代码进行说明；
 - (4) 主程序的设计：要求以适当的方式表达如何通过调用各个类及其成员函数来实现各项主要功能。
4. 调试分析：包括技术难点分析 和调试错误分析。
 - (1) 技术难点分析：要求至少说明一个或一个以上的技术难点。
 - (2) 调试错误分析：要求说明三个或三个以上的调度错误，及其修正方法。要求粘贴对应的调试截图，包括修正前的截图和修正后的截图。
5. 测试结果分析：
 - (1) 要求测试所有功能；
 - (2) 要求测试每个功能的各种可能的情况，包括边界数据、错误数据和其它可能的数据。
 - (3) 要求粘贴相应的测试截图，测试截图必须包含足够的提示信息和输出结果；
6. 附录：附上源代码，并标明源代码的所属文件，并且源代码必须有注释。

[实验报告排版要求]

1. 要求采用 WORD 进行文档撰写和排版；
2. 要求有封面，目录和页码，要求采用多级标题方式进行排版；
3. 正文字体建议为宋体，正文字体大小为小四或五号；正文行间距建议为 1.5 倍行间距；
4. 正文不要用黑体或彩色；粘贴的代码可以比正文字体小一号，代码及其注释要适当排版；
5. 实验报告中的图和表，要求进行编号。图和表的说明不要跨页，表格也尽量不要跨页；
6. 要求注意排版，排版的优劣将直接影响评分；

[提交内容]

1. 电子压缩包：包括实验报告(word 文档或 pdf 文档)和所有源代码(包括.h 文件和.cpp 文件)，采用 winra、winzip 或 7-zip 进行压缩。
2. 压缩文件名为：“学号+姓名”，如压缩文件“202109300619 杨露霞.rar”；如果是多人合作的，则压缩文件名为：“负责人学号+负责人姓名+参与者 1 学号+参与者 1 姓名+参与者 2 学号+参与者 2 姓名”，如压缩文件“202109300619 杨某某 202109300620 李某某 202109300631 张某某.rar”。

[考核方式]

1. 以小组方式进行面试，教师提问，根据工作量、程序演示情况、回答问题情况和实验报告情况，进行综合评分。
2. 同一小组，最多只能有一个优秀。没有在教师指定时间内完成课程设计和答辩环节的学生，成绩判定为不合格。
3. 具体评分标准，由任课教师根据以上原则制定。课题中的评分标准仅与程序实现相关，与报告、答辩表现的评分无关。

附：考核方式及评价标准

考核环节	考核方式	评分标准					占总成绩比例
		优	良	中	及格	不及格	
实践过程	教师和助教现场观察	对系统工作原理和实现过程理解透彻，有较好的见解和想法；实践过程操作及工具和使用非常娴熟；软件编程能力和调试能力强；遇到问题能够主动查阅资料，寻求各种途径去解决，并进行方法对比和筛选；积极参与实践全过程，团队协作性紧密，能够顾及同伴，在团队中贡献度大，责任感强；	对系统工作原理和实现过程理解比较深入；实践操作及工具和使用比较娴熟；软件设计能力和调试能力较好；遇到问题能主动解决，但是解决方法不够完善；积极参与实践全过程，团队内部比较团结，协作良好，能够顾及或帮助同伴，在团队中贡献度较大，有责任感。	对系统工作原理和实现过程理解一般，个别地方比较含糊；实践操作及工具和使用娴熟度一般，个别地方存在卡壳现象；软件设计能力和调试能力一般；遇到问题有主动解决的意识，但是还存在一些问题尚未解决思路；对团队事宜的参与度一般，不够积极，团队内部协作一般，不够紧密，存在成员孤立现象，对团队的贡献度一般，责任感一般。	对系统工作原理和实现过程理解比较不透彻，但有基本认知；实践操作及工具和使用不够熟悉，需要他人协助或查阅资料才能完成；软件设计能力和调试能力较差；遇到问题需要经常问老师或助教才能有解决思路，自己不能主动查阅资料解决；对团队事宜的参与度不积极，团队内部协作比较松散，成员之间沟通少或沟通不畅，存在成员孤立现象，对团队的贡献度低，责任感不强。	对系统工作原理和实现过程不清楚；不会操作相应的平台，不会使用相关工具和使用；软件编码能力不足；系统不能运行出正确结果或出现异常退出；遇到问题没有解决思路，也不主动寻求其它途径解决；缺席实践过程、结果演示、答辩和设计报告中任何一个环节；实践过程中进行其他与实践无关的事宜；团队协作参与度低，旁观他人操作，而自己动手很少；团队内部有较大分歧或争吵，不团结，不顾及他人，自私，没有责任感。	20%
系统结果演示	现场验收	参考题目后面的评价标准	参考题目后面的评价标准	参考题目后面的评价标准	参考题目后面的评价标准	系统使用往届或网络上已有的代码，算法理解差，系统功能少，结果不能正确解释；无法正确查找关键代码，算法总体思路和设计方法不清楚；演示时操作错误，系统异常退出，结果不正确。对系统出现的问题无法解释。	30%

答辩	教师当面考核	答辩时逻辑清晰，思路明确，语言流畅，陈述清楚，对实现过程和核心算法理解透彻，有独到的见解；回答问题快速准确。	答辩时逻辑比较清晰，语言陈述比较清楚，对实现过程和核心算法理解比较透彻；回答问题正确。	答辩时逻辑性一般，语言流畅度一般，陈述基本清楚，但陈述内容偏少；对实现过程和核心算法理解不够透彻；个别问题回答不全面。	答辩时逻辑性一般，语言流畅基本畅度，但表达不到位，陈述不够清楚，陈述内容少；对部分实现过程和核心算法理解不理解或理解不透彻；有一些问题回答不上来，或回答不全面。	答辩时逻辑性较差，内容比较混乱，语言表达不清，不知所云；陈述内容太少；对大部分实现过程和核心算法不理解或理解错误；有较多问题回答不上来或回答错误。	20%
设计报告	教师批改	设计报告结构清晰，内容完整，数据翔实；实现过程和关键技术分析描述深入、到位；调试结果正确，分析详细；源代码齐全；格式排版规范。	设计报告结构比较清晰，内容比较完整；实现过程和关键技术分析比较全面，但不够深入；调试结果正确，但分析不够详细；源代码齐全，但排版不够整齐；格式排版比较规范，个别地方存在小格式问题。	设计报告结构一般，有较多常识性内容，而核心实现过程和关键技术分析却偏少不够深入；内容比较完整；调试结果分析偏少，或者调试过程分析偏少；源代码不够齐全，排版也不够整齐；格式排版不够规范，有一些明显格式错误。	设计报告结构一般，常识性内容堆砌较多，而核心实现过程和关键技术分少；内容不够完整，缺少某一个部分；调试结果或调试过程分析少；只给出了部分源代码，且排版不够整齐；格式排版不规范，有较多明显格式错误。	设计报告逻辑性较差，结构比较混乱，缺少核心部分实现过程和关键技术；内容不完整，缺少调试结果或调试过程分析，或缺少源代码和个人感想等；排版混乱不规范，格式和内容错误多。	30%

1.图书馆书籍管理系统模拟

【问题描述】

在图书馆书籍管理系统中,管理员需要经常对图书信息进行查询和验证,如查询某本书是否在库、验证某本书的借阅状态等。由于图书信息的查询和验证频率很高,系统有必要有效地组织这些图书信息,从而快速查找和验证图书。另外,系统也会经常添加新入库的图书、删除损坏的图书和更新图书的借阅状态等操作,因此,系统必须采用动态结构,在添加、删除或更新后,依然能保证验证过程的快速。请采用相应的数据结构模拟图书馆书籍管理系统,其功能要求包括图书查询、图书状态更新、新书添加和旧书删除等。

【基本要求】

1. 要求自己编程实现二叉树结构及其相关功能,以存储图书信息,不允许使用标准模板类的二叉树结构和函数。同时要求根据二叉树的变化情况,进行相应的平衡操作,即 AVL 平衡树操作,四种平衡操作都必须考虑。测试时,各种情况都需要测试,并附上测试截图;
2. 要求采用类的设计思路,不允许出现类以外的函数定义,但允许友元函数。
3. 要求采用多文件方式: .h 文件存储类的声明, .cpp 文件存储类的实现,主函数 main 存储在另外一个单独的 cpp 文件中。如果采用类模板,则类的声明和实现都放在.h 文件中。
4. 不强制要求采用类模板;不强制要求采用可视化窗口,但若采用可视化窗口可适当提高考核分数;要求源程序中有相应注释;
5. 要求测试例子要比较详尽,各种极限情况也要考虑到,测试的输出信息要详细易懂,表明各个功能的执行正确;

【实现提示】

1. 图书信息(即图书编号和借阅状态)可以存储在文件中,当程序启动时,从文件中读取所有的图书信息,并建立合适的查找二叉树;
2. 查询过程时,需要根据图书的编号,检索整个二叉树,找到匹配的图书编号,进行验证;更新图书借阅状态时,也需要检索二叉树,找到匹配项后进行更新,同时更新文件中存储的图书借阅状态;
3. 添加图书时,不仅需要在文件中添加,也需要在二叉树中添加相应的节点;删除图书时,也是如此;
4. 图书馆书籍管理系统可以联系实际,扩展相应功能。

【运行结果要求】

要求有程序菜单,能够实现**用户的登录管理、图书查询、添加图书、删除图书和更新图书借阅状态功能**,实验报告要求有详细的功能测试截图。

程序评价标准:

优秀: 在良好的基础上,联系实际系统应用情况,增加各种扩展功能。

良好: 在中等的基础上,菜单设计逻辑清晰、美观。能进行用户管理及 AVL 相关功能的可视化展现。

中等: 在及格基础上,设计良好的程序菜单,逻辑正常,功能可正常跳转。

及格: 实现了 AVL 树类,图书类等基本结构,能进行一定的基本操作,包括查询、增加删除等。

2 地图网络的图分析

【问题描述】

地图网络可以被看作是一个图，其中节点代表城市或地区，边代表城市或地区之间的交通路线。请设计一个程序来模拟地图网络，并使用图来表示这个网络。系统需要有用户登录功能，用户能够查看整个地图网络信息，添加/删除节点和边，查找特定节点的所有邻近节点，以及查找两个节点之间的最短路径，用于点到点的导航。

【基本要求】

1. 自己编程实现图结构，包括邻接矩阵与邻接表，及其相关功能，不允许使用标准模板类的图结构和函数。同时要求根据图的变化情况，进行相应的图遍历操作，包括但不限于深度优先搜索和广度优先搜索。测试时，各种情况都需要测试，并附上测试截图；
2. 要求采用类的设计思路，不允许出现类以外的函数定义，但允许友元函数。
3. 要求采用多文件方式：.h 文件存储类的声明，.cpp 文件存储类的实现，主函数 main 存储在另外一个单独的 cpp 文件中。如果采用类模板，则类的声明和实现都放在.h 文件中。
4. 不强制要求采用类模板；不强制要求采用可视化窗口，但若采用可视化窗口可适当提高考核分数；要求源程序中有相应注释；
5. 要求测试例子要比较详尽，各种极限情况也要考虑到，测试的输出信息要详细易懂，表明各个功能的执行正确；

【实现提示】

1. 城市或地区信息(即城市或地区名称和交通路线)可以存储在文件中，当程序启动时，从文件中读取所有的城市或地区信息，并建立合适的图结构；
2. 验证过程时，需要根据城市或地区名称，遍历整个图，找到匹配的城市或地区，进行验证；更新交通路线时，也需要遍历图，找到匹配项后进行更新，同时更新文件中存储的交通路线。
3. 路径的搜索实现可以利用栈或队列结构
4. 添加城市或地区时，不仅需要在文件中添加，也需要在图中添加相应的节点；删除城市或地区时，也是如此；
5. 地图网络系统可以联系实际，扩展相应功能。例如，实现地图网络的可视化等。

【运行结果要求】

要求有程序菜单，能够实现用户的登录管理、地区信息增删改查、地区间导航功能及相应的信息展示，实验报告要求有详细的功能测试截图。

程序评价标准：

优秀：在良好的基础上，联系实际情况，增加各种扩展能。

良好：在中等的基础上，菜单设计逻辑清晰、美观，并完整实现最短路径算法。

中等：在及格基础上，实现用户管理功能，以及良好的信息展示界面，如地图网络或地区信息。

及格：实现了图类，城市或地区类等基本结构，能进行一定的基本操作，包括地区（及地区之间）信息查询、增加删除等。实现基本的点到点路径搜索。

3 文件压缩工具的模拟

【问题描述】

哈夫曼编码是一种有效的文件压缩方法,它通过为文件中的每种字符分配一个唯一的二进制编码来工作。编码的长度与字符的频率成反比,即频率高的字符有更短的编码,频率低的字符有更长的编码。请设计一个程序来模拟这个过程。

首先,程序应该能够读取一个文本文件,并计算文件中每种字符的频率。然后,根据这些频率,程序应生成一个哈夫曼编码树,并使用这个树来为每种字符生成其对应的哈夫曼编码。接下来,程序应该使用这些编码来生成一个压缩的版本的输入文件。

此外,程序还应该能够进行解压缩操作。它应该能够读取一个压缩的文件,然后使用哈夫曼编码树来解码文件中的每个字符,从而恢复原始的文本。

【基本要求】

1.自己编程实现哈夫曼树及其相关功能,要求根据字符频率的变化情况,动态地生成和调整哈夫曼树。测试时,各种情况都需要测试,并附上测试截图;

2.要求采用类的设计思路,不允许出现类以外的函数定义,但允许友元函数。

3.要求采用多文件方式: .h 文件存储类的声明, .cpp 文件存储类的实现,主函数 main 存储在另外一个单独的 cpp 文件中。如果采用类模板,则类的声明和实现都放在.h 文件中。

4.不强制要求采用类模板;不强制要求采用可视化窗口,但若采用可视化窗口可适当提高考核分数;要求源程序中有相应注释;

5.要求测试例子要比较详尽,各种极限情况也要考虑到,测试的输出信息要详细易懂,表明各个功能的执行正确;

【实现提示】

1.文件的读取和写入可以使用标准库中的文件操作函数。读取时,可以一次读取一个字符,计算字符的频率;写入时,可以一次写入一个字符,或者写入哈夫曼编码。

2.哈夫曼编码的生成需要根据字符的频率来进行。频率高的字符的编码长度应该短,频率低的字符的编码长度应该长。可以使用优先队列来帮助生成哈夫曼树。

3.压缩文件和解压缩文件时,需要分别生成和解析哈夫曼编码。可以在哈夫曼树中添加相应的操作来完成这些任务。

【运行结果要求】

要求有程序菜单,能够实现文件的读写、哈夫曼树及其编码构建、信息展示、文件压缩与解压功能,实验报告要求有详细的功能测试截图。

程序评价标准:

优秀: 在良好的基础上,实现文件的管理。并联系实际情况,增加各种扩展能。

良好: 在中等的基础上,菜单设计逻辑清晰、美观,实现了文件的压缩功能,能够对读取的文件内容进行哈夫曼编码,并将编码结果写入压缩文件。同时实现了文件的解压缩功能,能够读取压缩文件,进行哈夫曼解码,并将解码结果恢复为原始文件内容。

中等: 在及格的基础上,拥有基本的菜单设计,并通过控制台输出相应编码过程信息。

及格: 实现了基础的哈夫曼树构造和编码生成,能够对给定的字符进行编码和解码。

4 排序算法的控制台可视化程序

【问题描述】

设计一个程序来可视化各种排序算法的执行过程。这个程序应该能够生成一组随机的数字，然后使用多种排序算法（如冒泡排序、插入排序、选择排序、快速排序、归并排序等）对它们进行排序。在排序过程中，程序应该在控制台上以动态的方式显示排序的每一步，以便用户可以看到排序算法是如何一步步地将数字排序的。同时，程序应该提供一种方式，让用户可以选择不同的排序算法，并可以控制排序过程的播放速度。例如，用户可以选择以慢速或快速的方式播放排序过程。

【基本要求】

1. 自己编程实现多种排序算法，不允许使用标准模板类的排序函数。测试时，各种情况都需要测试，并附上测试截图。要求采用类的设计思路，不允许出现类以外的函数定义，但允许友元函数。
2. 设计图形用户界面，可以清楚地显示排序过程中数组的状态变化。用户应该能够控制排序过程的播放速度
3. 实现随机数生成功能，在不需要进行可视化时，要求排序数据量大于 10000。并可输出排序信息，包括运行时间、比较次数、交换次数、内存使用量、稳定性、函数调用次数、循环次数和嵌套深度信息，并且可以按照不同的指标对排序算法的优劣进行排名。
4. 要求采用多文件方式：.h 文件存储类的声明，.cpp 文件存储类的实现，主函数 main 存储在另外一个单独的 cpp 文件中。如果采用类模板，则类的声明和实现都放在.h 文件中。
5. 要求测试例子要比较详尽，各种极限情况也要考虑到，测试的输出信息要详细易懂，表明各个功能的执行正确。

【实现提示】

1 考虑到控制台单屏输出内容有限，当选择进行排序可视化展示时，可以将排序数据控制在 20 个以内。

2 可以使用数组或向量来存储要排序的数字。在进行排序时，需要在每一步都更新显示，以反映数组的当前状态；

3 在控制台上显示排序过程可能会比较复杂，同学们可以自行灵活设计。一种可行的方法是在每一行上打印出一个表示数字大小的条形图。例如，如果一个数字的值是 10，那么就打印出 10 个星号或其他字符。每一行代表一个数字，每次输出当前排序步骤的结果后，清屏再输出下一步骤结果

4 对于控制排序过程的播放速度，一种可能的方法是在每一步排序后都暂停一定的时间。这个时间可以由用户通过输入来调整。

【运行结果要求】

要求有程序菜单，能够用类的方式实现**课堂教学的基本排序算法、随机数生成功能、排序算法可视化动画**，实验报告要求有详细的功能测试截图。

程序评价标准：

优秀：在良好的基础上，实现复合型排序算法，能够根据数据规律选择合适的排序方法。并联系实际情况，增加各种扩展能。

良好：在中等的基础上，实现完善的随机数据生成功能、额外 2-4 种排序算法，这些额外实现也可以是针对某一算法的优化版本。

中等：在及格的基础上，拥有基本的菜单设计，拥有排序可视化的基本展示方案。

及格：使用面向对象编程，实现基本的 8 种排序算法，并能统计算法的基本性能信息，打印基本的排序过程信息。

5 浏览器历史记录管理模拟

【问题描述】

设计一个浏览器历史记录管理模拟程序。这个程序应该能够记录用户浏览的网页，并提供“后退”、“前进”、“清除历史记录”和“查找历史记录”等功能。每当用户浏览一个新的网页，就将其添加到历史记录中；每当用户点击“后退”按钮，就返回到上一个网页；每当用户点击“前进”按钮，就返回到下一个网页；每当用户点击“清除历史记录”按钮，就清空所有的历史记录；每当用户输入关键字进行查找，就返回包含这个关键字的所有历史记录。

【基本要求】

本题可涉及树 or 图（用于网页及链接的数据管理）、栈（用于管理历史记录），请编程实现结构及其相关功能，

对于网页信息，提前生成模拟数据保存在文件中，方便程序载入时读取，要求总体网页数据量大于 100 条。

测试时，各种情况都需要测试，并附上测试截图；

要求采用类的设计思路，不允许出现类以外的函数定义，但允许友元函数。

要求测试例子要比较详尽，各种极限情况也要考虑到，测试的输出信息要详细易懂，表明各个功能的执行正确；

【实现提示】

1.网页信息应包含网页编号、网页标签、转向其它页面的 URL、网页内容摘要等信息。当程序启动时，从文件读取所有网页信息，并构建相应的树或图结构。

2.开始运行时，可模拟用户浏览网页流程，给予用户当前的页面信息以及可以浏览的 URL，用户可根据 URL 标签进一步浏览信息。

3.系统应可以记录用户的浏览历史，用户可以随时后退或前进进行网页浏览，当已浏览过的网页的为空时，不能执行后退操作。当可以前进到的网页为空时，不能执行前进操作。当执行访问其它网页操作时，需要清空可以前进到的网页。

4.系统可以通过关键字查询相应的浏览历史。

5.浏览记录系统可以联系实际，扩展相应功能。例如，实现浏览顺序的可视化等。

【运行结果要求】

要求有程序菜单，能够用类的方式实现网页类、树或图类，可以进行用户管理，进行网页信息的增删改查，网页浏览的前进后退操作，输出网页内容信息及网站整体网页拓扑信息，实验报告要求有详细的功能测试截图。

程序评价标准：

优秀：在良好的基础上，在良好的基础上，实现更高级的功能，如历史记录的可视化、网页浏览路径的推荐等。

良好：在中等的基础上，实现完善的用户访问，前进后退功能，可输出浏览历史、清空历史。并提供美观的控制台菜单及信息可视化设计。

中等：在及格的基础上，拥有基本的菜单设计，从用户角度对网页进行浏览。拥有事先生成好的大量网页数据，程序直接从文件读取。有基本的前进和后退功能。

及格：使用面向对象编程，实现网页类及相应的存储数据结构类，可以对网页进行访问，对信息进行增加或删除，可利用关键字进行网页查找。

6 简易区块链模拟

【问题描述】

区块链是一种分布式数据库技术，广泛应用于比特币和其他虚拟货币中。在区块链中，每个“区块”包含一组交易记录，并且链接到前一个区块，形成一个链条。每个区块都包含前一个区块的哈希值，这使得更改历史记录变得非常困难。请设计一个程序来模拟区块链的基本操作，并使用哈希表来存储区块链数据。

这个程序应该能够添加新的区块，每个区块包含一组交易数据和前一个区块的哈希值（该值由前区块的所有内容计算得到）。程序还应该能够验证区块链的完整性，即检查每个区块的哈希值是否与其内容和前一个区块的哈希值匹配。

【基本要求】

自己编程实现哈希表及其相关功能，如插入、查找等，不允许使用标准模板类的哈希表结构和函数。测试时，各种情况都需要测试，并附上测试截图；

要求采用类的设计思路，不允许出现类以外的函数定义，但允许友元函数。

要求测试例子要比较详尽，各种极限情况也要考虑到，测试的输出信息要详细易懂，表明各个功能的执行正确。

【实现提示】

1. 区块链可以存储在文件中，当程序启动时，从文件中读取所有的区块，并建立哈希表；
2. 在实现这个问题时，我们可以将每个区块看作是链表的一个节点，节点中包含该区块的哈希值，该区块的交易数据，以及指向前一个区块（即前一个节点）的指针。这样，我们就可以使用链表来表示整个区块链。同时，为了能够快速查找和验证一个区块，我们可以使用哈希表来存储区块链。
3. 在添加新的区块时，需要计算新区块的哈希值，并在哈希表中插入新的键值对。
4. 在验证区块链的完整性时，需要遍历哈希表，检查每个区块的哈希值是否与其内容和前一个区块的哈希值匹配。

【运行结果要求】

要求有程序菜单，能够用类的方式实现**区块链类、哈希表类**，可以进行**区块的添加，数据的验证功能**，实验报告要求有详细的功能测试截图。

程序评价标准：

本题最高评分为良好。

良好：在中等的基础上，拥有详细的控制台信息输出，对新建区块、查找区块、验证区块流程有详细的解释。

中等：在及格的基础上，拥有基本的菜单设计，通过文件读写，提前建立数据量足够大的区块链，设计哈希算法，并同步哈希表数据。

及格：使用面向对象编程，实现区块链类、哈希表类，可进行一定程度的数据验证。

7 大整数的运算

【问题描述】

密码学分为两类密码：对称密码和非对称密码。对称密码主要用于数据的加/解密，而非对称密码则主要用于认证、数字签名等场合。非对称密码在加密和解密时，是把加密的数据当作一个大的正整数来处理，这样就涉及到大整数的加、减、乘、除和指数运算等，同时，还需要对大整数进行输出。请采用相应的数据结构实现大整数的加、减、乘、除和指数运算，以及大整数的输入和输出。

【基本要求】

1. 要求采用链表来实现大整数的存储和运算，不允许使用标准模板类的链表类(list)和函数。同时要求可以从键盘输入大整数，也可以文件输入大整数，大整数可以输出至显示器，也可以输出至文件。大整数的存储、运算和显示，可以同时支持二进制和十进制，但至少支持十进制。大整数输出显示时，必须能清楚地表达出整数的位数。测试时，各种情况都需要测试，并附上测试截图；要求测试例子要比较详尽，各种极限情况也要考虑到，测试的输出信息要详细易懂，表明各个功能的执行正确；
2. 要求大整数的长度可以不受限制，即大整数的十进制位数不受限制，可以为十几位的整数，也可以为 500 多位的整数，甚至更长；大整数的运算和显示时，只需要考虑正的大整数。如果可能的话，请以秒为单位显示每次大整数运算的时间；
3. 要求采用类的设计思路，不允许出现类以外的函数定义，但允许友元函数。主函数中只能出现类的成员函数的调用，不允许出现对其它函数的调用。
4. 要求采用多文件方式：.h 文件存储类的声明，.cpp 文件存储类的实现，主函数 main 存储在另外一个单独的 cpp 文件中。如果采用类模板，则类的声明和实现都放在.h 文件中。
5. 不强制要求采用类模板，也不要要求采用可视化窗口；要求源程序中有相应注释；

【实现提示】

1. 大整数的加减运算可以分解为普通整数的运算来实现；而大整数的乘、除和指数运算，可以分解为大整数的加减运算。
2. 大整数的加、减、乘、除和指数运算，一般是在求两大整数在取余操作下的加、减、乘、除和指数运算，即分别求 $(a + b) \bmod n$, $(a - b) \bmod n$, $(a * b) \bmod n$, $(a / b) \bmod n$ 和 $(a^b) \bmod n$ 。其中 a^b 是求 a 的 b 次方，而 n 称之为模数。说明：取余操作(即 mod 操作)是计算相除之后所得的余数，不同于除法运算的是，取余操作得到的是余数，而不是除数。如 $7 \bmod 5 = 2$ 。模数 n 的设定，可以为 2^m 或 10^m , m 允许每次计算时从键盘输入。模数 n 的取值一般为 2512(相当于十进制 150 位左右),21024(相当于十进制 200~300 位),22048(相当于十进制 300~500 位)。为了测试，模数 n 也可以为 2256, 2128 等值。
3. 需要设计主要类有：链表类和大整数类。链表类用于处理链表的相关操作，包括缺省构造函数、拷贝构造函数、赋值函数、析构函数、链表的创建、插入、删除和显示等；而大整数类则用于处理大整数的各种运算和显示等。

【运行结果要求】

要求能实现大整数的加、减、乘、除和指数运算，以及大整数的输入和输出，实验报告要求有详细的设计思路、功能测试截图。

程序评价标准：

本题最高评分为良好，除非对优化算法有深入研究，并给出完善的讨论和实现方案。(优秀：对大整数的存储设计进行优化思考和实现，对乘法、乘方和除法的算法有深入的研究讨论，完善考虑优化方案，给出完整的设计思路，并实现优化后的策略，对实现策略有相应的评估和论证。)

良好：在中等的基础上，有文件数据读写的功能，测试用例完善。能在合理时间内对复杂的计算输出结果。

中等：在及格的基础上，拥有基本的菜单设计，逻辑跳转正常。

及格：使用面向对象编程，实现大整数类，能够进行基本的加减乘除操作。