

# 2023年大数据课程总结(优秀9篇)

总结是写给人看的，条理不清，人们就看不下去，即使看了也不知其所以然，这样就达不到总结的目的。大家想知道怎么样才能写一篇比较优质的总结吗？下面是小编带来的优秀总结范文，希望大家能够喜欢！

## 大数据课程总结篇一

会计课程作为管理学的一门重要课程，在大学教育中占据着重要的位置。作为一名大学生，我在学习会计课程时，不仅仅掌握了一些基本的会计知识和技能，更重要的是学到了一些关于思考和解决问题的方法。在这篇文章中，我将总结并分享我对会计课程设计的心得体会，希望能对大家有所启发。

首先，在会计课程的设计中，理论与实践的结合十分重要。会计并不是一个纯粹的理论学科，而是需要在实际操作中应用的技能。因此，好的会计课程设计应该将理论知识与实践相结合，让学生们能够通过案例分析和实际操作来巩固所学的知识。在我的学习过程中，我们经常通过实验室和实操课来进行会计报表的编制和分析，这让我更加深入地理解了会计的核心概念和方法。

其次，提供实际问题的案例研究是培养学生解决问题能力的重要途径。在会计课程中，不仅要学习会计准则和规范，还要具备处理实际问题的能力。因此，设计师应该提供一些真实的会计案例，让学生们通过分析、思考和解决问题来应用所学知识。通过与同学们的讨论和辩论，我们能够从不同的角度思考问题，并找到最佳的解决方案。这种案例学习的方式让我感受到了解决问题的乐趣，并提升了我的批判性思维能力。

再次，培养团队合作和沟通能力是会计课程设计的重要目标

之一。在现实工作中，一个会计师往往需要与很多其他部门的人员进行合作和沟通。因此，设计师应该通过小组项目和课堂讨论等方式，培养学生们的团队合作和沟通能力。在我的会计课程中，我们经常以小组为单位完成一些会计实践项目。在这个过程中，我明白了团队合作的重要性，并学会了如何与他人有效地沟通和合作。这对我来说是一种宝贵的经验，将来我将能够在工作中更好地与他人合作。

另外，会计课程的设计应该注重学生的职业发展和素养培养。作为一门专业课程，会计课程应该向学生传授一些与职业发展和素养相关的知识。例如，会计伦理、职业道德、和讲述一些成功会计师的故事等。这些内容能够让学生们了解自己的职业道路，并提升他们的专业素养。在我的学习中，我们学习了会计伦理和职业道德的相关知识，这让我明白了作为一名会计师应该如何行事，并且有助于我树立正确的职业价值观。

最后，会计课程设计的目标应该是培养学生的终身学习能力。在这个快速变革的时代，会计准则和法规不断更新，专业知识也在不断增加。因此，好的会计课程设计应该培养学生的终身学习能力，使他们能够适应未来的变化。在我的会计课程中，我们被鼓励通过课后作业、自主学习和实践等方式来不断巩固和拓宽自己的知识。这种学习方式让我明白了学习的过程不仅仅是为了应付考试，更是为了成长和发展自己。

总而言之，好的会计课程设计应该注重理论与实践的结合，提供实际问题的案例研究，培养团队合作和沟通能力，注重学生的职业发展和素养培养，并培养学生的终身学习能力。通过我在会计课程中的学习和实践，我深刻体会到了这些设计原则的重要性，并相信这些体会将对我的未来发展产生积极的影响。作为一名学生，我将更加努力学习，不断提升自己的会计技能和综合素质，为将来的职业生涯做好充分的准备。

## 大数据课程总结篇二

《数据结构》是一门实践性较强的课程，为了学好这门课程，必须在掌握理论知识的同时，加强上机实践。

接下来就跟本站小编一起去了解一下关于数据结构课程设计心得体会吧！

决问题和在老师的帮助下一步一步慢慢的正确运行程序，决问题和在老师的帮助下一步一步慢慢的正确运行程序，终于完成了这次课程设计，于完成了这次课程设计，虽然这次课程设计结束了但是总觉得自己懂得的知识很是不足，学无止境，得自己懂得的知识很是不足，学无止境，以后还会更加的努力深入的学习。力深入的学习。

本次课程设计，使我对《数据结构》这门课程有了更深入的理解。我的课程设计题目是线索二叉树的运算。刚开始做这个程序的时候，感到完全无从下手，甚至让我觉得完成这次程序设计根本就是不可能的，于是开始查阅各种资料以及参考文献，之后便开始着手写程序，写完运行时有很多问题。特别是实现线索二叉树的删除运算时很多情况没有考虑周全，经常运行出现错误，但通过同学间的帮助最终基本解决问题。

在本课程设计中，我明白了理论与实际应用相结合的重要性，并提高了自己组织数据及编写大型程序的能力。培养了基本的、良好的程序设计技能以及合作能力。这次课程设计同样提高了我的综合运用所学知识的能力。并对VC有了更深入的了解。《数据结构》是一门实践性很强的课程，上机实习是对学生全面综合素质进行训练的一种最基本的方法，是与课堂听讲、自学和练习相辅相成的、必不可少的一个教学环节。上机实习一方面能使书本上的知识变“活”，起到深化理解和灵活掌握教学内容的目的；另一方面，上机实习是对学生软件设计的综合能力的训练，包括问题分析，总体结构设计，程序设计基本技能和技巧的训练。此外，还有更重要的一点

是：机器是比任何教师更严厉的检查者。因此，在“数据结构”的学习过程中，必须严格按照老师的要求，主动地、积极地、认真地做好每一个实验，以不断提高自己的编程能力与专业素质。

通过这段时间的课程设计，我认识到数据结构是一门比较难的课程。需要多花时间上机练习。这次的程序训练培养了我实际分析问题、编程和动手能力，使我掌握了程序设计的基本技能，提高了我适应实际，实践编程的能力。

总的来说，这次课程设计让我获益匪浅，对数据结构也有了进一步的理解和认识。

通过本次课程设计，对图的概念有了一个新的认识，在学习离散数学的时候，总觉得图是很抽象的东西，但是在学习了《数据结构与算法》这门课程之后，我慢慢地体会到了其中的奥妙，图能够在计算机中存在，首先要捕捉他有哪些具体化、数字化的信息，比如说权值、顶点个数等，这也就说明了想要把生活中的信息转化到计算机中必须用数字来完整的构成一个信息库，而图的存在，又涉及到了顶点之间的联系。图分为有向图和无向图，而无向图又是有向图在权值双向相等下的一种特例，如何能在计算机中表示一个双向权值不同的图，这就是一件很巧妙的事情，经过了思考和老师同学的帮助，我用 `edges[i][j]=up` 和 `edges[j][i]=up` 就能实现了一个双向图信息的存储。对整个程序而言，`dijkstra` 算法始终都是核心内容，其实这个算法在实际思考中并不难，也许我们谁都知道找一个路径最短的方法，及从顶点一步一步找最近的路线并与其直接距离相比较，但是，在计算机中实现这么一个很简单的想法就需要涉及到很多专业知识，为了完成设计，在前期工作中，基本都是以学习C语言为主，所以浪费了很多时间，比如说在程序中，删除顶点和增加顶点的模块中都有和建图模块相互重复的函数，但是由于技术的原因，只能做一些很累赘的函数，可见在调用知识点，我没有掌握好。不过，有了这次课程设计的经验和教训，我能够很清楚

的对自己定一个合适的水平，而且在这次课程设计中我学会了运用两个新的函数printf和包涵在#include头文件中的输入函数。因为课程设计的题目是求最短路径，本来是想通过算法的实现把这个程序与交通情况相连，但是因为来不及查找各地的信息，所以，这个计划就没有实现，我相信在以后有更长时间的情况下，我会做出来的。

## 大数据课程总结篇三

本次课程设计，使我对《数据结构》这门课程有了更深入的理解。《数据结构》是一门实践性较强的课程，为了学好这门课程，必须在掌握理论知识的同时，加强上机实践。

我的课程设计题目是线索二叉树的运算。刚开始做这个程序的时候，感到完全无从下手，甚至让我觉得完成这次程序设计根本就是不可能的，于是开始查阅各种资料以及参考文献，之后便开始着手写程序，写完运行时有很多问题。特别是实现线索二叉树的删除运算时很多情况没有考虑周全，经常运行出现错误，但通过同学间的帮助最终基本解决问题。

在本课程设计中，我明白了理论与实际应用相结合的重要性，并提高了自己组织数据及编写大型程序的能力。培养了基本的、良好的程序设计技能以及合作能力。这次课程设计同样提高了我的综合运用所学知识的能力。并对vc有了更深入的了解。《数据结构》是一门实践性很强的课程，上机实习是对学生全面综合素质进行训练的一种最基本的方法，是与课堂听讲、自学和练习相辅相成的、必不可少的一个教学环节。上机实习一方面能使书本上的知识变“活”，起到深化理解和灵活掌握教学内容的目的；另一方面，上机实习是对学生软件设计的综合能力的训练，包括问题分析，总体结构设计，程序设计基本技能和技巧的训练。此外，还有更重要的一点是：机器是比任何教师更严厉的检查者。因此，在“数据结构”的学习过程中，必须严格按照老师的要求，主动地、积

极地、认真地做好每一个实验，以不断提高自己的编程能力与专业素质。

通过这段时间的课程设计，我认识到数据结构是一门比较难的课程。需要多花时间上机练习。这次的程序训练培养了我实际分析问题、编程和动手能力，使我掌握了程序设计的基本技能，提高了我适应实际，实践编程的能力。

## 大数据课程总结篇四

数据结构是计算机程序设计的重要理论技术基础，它不仅是计算机科学的核心课程，而且也已经成为其他理工专业的热门选修课。随着高级语言的发展，数据结构在计算机的研究和应用中已展现出强大的生命力，它兼顾了诸多高级语言的特点，是一种典型的结构化程序设计语言，它处理能力强，使用灵活方便，应用面广，具有良好的可移植性。

紧张的两周数据结构实训很快就过去了，通过这两周的实践学习，不仅使我们巩固了以前的知识并在此基础上还对数据结构的特点和算法有了更深的了解，使我们在这门课程的实际应用上也有了一个提高。

首先这两周的学习，使我们在巩固了原有的理论知识上，又培养了灵活运用和组成所学过知识及技能来分析、解决实际问题的能力，使我们体会到自身知识和能力在实际中的应用和发挥。

其次，它激发了我们创新意识，开发创造的能力和培养沟通能力。另外，让我们进一步熟悉了数据结构的设计应用。每一处编码都是在反复的熟悉数据结构的结构特性，及其语法、函数和程序设计思想的过程，对我们数据结构的学习和提高很有益处，并且使我们明白了程序设计过程，如解决一些实际问题，从解决实际问题的角度，我们可以这样来看：

第一要了解这个问题的基本要求，即输入、输出、完成从输入到输出的要求是什么；

第二，从问题的要害入手，从前到后的解决问题的每个方面，即从输入开始入手，着重考虑如何从输入导出输出，在这个过程中，可确定所需的数据结构的基本类型——线性表、栈、队列、串、数组、广义表、树和二叉树以及图等，然后确定处理过程——算法，通过在编译环境中的编译与调试，可到最终的程序。最后，在这次的实训过程中，我们深刻的认识到了自己在学习方面的不足之处，我知道我还有太多的基本的思想没有真正的理解，当然我们不会灰心，我们会在以后的日子里努力弥补我们的不足。

在两周的实训中，我们也体会到了团队合作的重要性，从最初的查阅资料到最后的程序的成功运行，我们组有过山穷水尽的困惑；有过柳暗花明的惊喜；有过唇枪舌剑的辩论；有过相互鼓励的安慰。两个礼拜的时间我们经历了很多，也收获了很多。与其说这次的实训是体力与脑力的作业，不如说它是合作精神和毅力的考验。经过这次课程设计，我们不仅学到了很多知识和技能，更重要的是我们学会了如何运用所学知识去解决实际问题。

总之，两个礼拜的课程设计让我们受益匪浅。我们深深认识到，要学好一门学科，没有刻苦钻研的精神是不行的，只有在不断的尝试中，经历失败，从失败中经验，然后再不断的尝试，才能获得成功。

## 大数据课程总结篇五

本次课程设计，使我对《数据结构》这门课程有了更深入的理解。《数据结构》是一门实践性较强的课程，为了学好这门课程，必须在掌握理论知识的同时，加强上机实践。

我的课程设计题目是线索二叉树的运算。刚开始做这个程序

的时候，感到完全无从下手，甚至让我觉得完成这次程序设计根本就是不可能的，于是开始查阅各种资料以及参考文献，之后便开始着手写程序，写完运行时有很多问题。特别是实现线索二叉树的删除运算时很多情况没有考虑周全，经常运行出现错误，但通过同学间的帮助最终基本解决问题。

在本课程设计中，我明白了理论与实际应用相结合的重要性，并提高了自己组织数据及编写大型程序的能力。培养了基本的、良好的程序设计技能以及合作能力。这次课程设计同样提高了我的综合运用所学知识的能力。并对VC有了更深入的了解。《数据结构》是一门实践性很强的课程，上机实习是对学生全面综合素质进行训练的一种最基本的方法，是与课堂听讲、自学和练习相辅相成的、必不可少的一个教学环节。上机实习一方面能使书本上的知识变“活”，起到深化理解和灵活掌握教学内容的目的；另一方面，上机实习是对学生软件设计的综合能力的训练，包括问题分析，总体结构设计，程序设计基本技能和技巧的训练。此外，还有更重要的一点是：机器是比任何教师更严厉的检查者。因此，在“数据结构”的学习过程中，必须严格按照老师的要求，主动地、积极地、认真地做好每一个实验，以不断提高自己的编程能力与专业素质。

通过这段时间的课程设计，我认识到数据结构是一门比较难的课程。需要多花时间上机练习。这次的程序训练培养了我实际分析问题、编程和动手能力，使我掌握了程序设计的基本技能，提高了我适应实际，实践编程的能力。

总的来说，这次课程设计让我获益匪浅，对数据结构也有了进一步的理解和认

## 大数据课程总结篇六

本文目录



1. 数据库课程设计心得体会
2. 数据库课程设计心得体会

两个星期的时间非常快就过去了，这两个星期不敢说自己有多大的进步，获得了多少知识，但起码是了解了项目开发的部分过程。虽说上过数据库上过管理信息系统等相关的课程，但是没有亲身经历过相关的设计工作细节。这次实习证实提供了一个很好的机会。

通过这次课程设计发现这其中需要的很多知识我们没有接触过，去图书馆查资料的时候发现我们前边所学到的仅仅是皮毛，还有很多需要我们去掌握的东西我们根本不知道。同时也发现有很多已经学过的东西我们没有理解到位，不能灵活运用于实际，不能很好的用来解决问题，这就需要我们不断的大量的实践，通过不断的自学，不断地发现问题，思考问题，进而解决问题。在这个过程中我们将深刻理解所学知识，同时也可以学到不少很实用的东西。

从各种文档的阅读到开始的需求分析、概念结构设计、逻辑结构设计、物理结构设计。亲身体验了一回系统的设计开发过程。很多东西书上写的很清楚，貌似看着也很简单，思路非常清晰。但真正需要自己想办法去设计一个系统的时候才发现其中的难度。经常做到后面突然就发现自己一开始的设计有问题，然后又回去翻工，在各种反复中不断完善自己的想法。

我想有这样的问題不止我一个，事后想想是一开始着手做的时候下手过于轻快，或者说是根本不了解自己要做的这个系统是给谁用的。因为没有事先做过仔细的用户调查，不知道整个业务的流程，也不知道用户需要什么功能就忙着开发，这是作为设计开发人员需要特别警惕避免的，不然会给后来的工作带来很大的麻烦，甚至可能会需要全盘推倒重来。所以以后的课程设计要特别注意这一块的设计。

按照要求，我们做的是机票预订系统。说实话，我对这个是一无所知的，没有订过机票，也不知道航空公司是怎么一个流程。盲目开始设计的下场我已经尝过了，结果就是出来一个四不像的设计方案，没有什么实际用处。没有前期的调查，仅从指导书上那几条要求着手是不够的。

在需求分析过程中，我们通过上网查资料，去图书馆查阅相关资料，结合我们的生活经验，根据可行性研究的结果和客户的要求，分析现有情况及问题，采用client/server结构，将机票预定系统划分为两个子系统：客户端子系统，服务器端子系统。在两周的时间里，不断地对程序及各模块进行修改、编译、调试、运行，其间遇到很多问题：由于忘记了一些java语言的规范使得在调试过程中一些错误没有发现，通过这次课程设计，我对调试掌握得更加熟练了，意识到了程序语言的规范性以及我们在编程时要有严谨的态度，同时在写程序时如有一定量的注释，既增加了程序的可读性，也可以使自己在读程序时更容易。

我们学习并应用了sql语言，对数据库的创建、修改、删除方法有了一定的了解，通过导入表和删除表、更改表学会了对于表的一些操作，为了建立一个关系数据库信息管理系统，必须得经过系统调研、需求分析、概念设计、逻辑设计、物理设计、系统调试、维护以及系统评价的一般过程，为毕业设计打下基础。

很多事情不是想象中的那么简单的，它涉及到的各种实体、属性、数据流、数据处理等等。很多时候感觉后面的设计根本无法继续，感觉像是被前面做的各种图限制了。在做关系模型转换的时候碰到有些实体即可以认为是实体又可以作为属性，为了避免冗余，尽量按照属性处理了。

物理结构设计基本没有碰到问题，这一块和安全性、完整性不觉就会在物理结构设计中添加一些安全设置：主键约束、check约束、default定义等。最后才做索引的部分，对一

些比较经常使用搜索的列，外键上建立索引，这样可以明显加快检索的速度，最后别忘记重要的安全性设置，限制用户访问权限，新建用户并和数据库用户做相应的映射。

不管做什么，我们都要相信自己，不能畏惧，不能怕遇到困难，什么都需要去尝试，有些你开始认为很难的事在你尝试之后你可能会发现原来她并没有你以前觉得的那样，自己也是可以的。如果没有自信，没有目标，没有信心就不可能把事情做好，当其他人都在迷茫的时候，自己一定要坚信目标，大学毕业出去即面临找工作，从学习这个专业，到以后从事这方面的工作都需要不断地去学习去实践，这次实践可以给我们敲一个警钟，我们面临毕业，面临择业，需要这些实践经验，在困难面前要勇于尝试，这是这次课程设计给我的最大感想！

以上基本是这次实习的体会了，设计进行的非常艰难，编码非常不容易，才发现做一个项目最重要的不在于如何实现，而是实现之前的需求分析和模块设计。创新很难，有些流行的系统其实现并不难，难的在于对市场的分析和准确定位。设计，是一个任重道远的过程。

数据库课程设计心得体会2篇 | [返回目录](#)

首先，让我的记忆追溯到大二暑假，在老大的指引下（），我接触到microsoft 产品。那个时候我已经学过vc和asp[]因为windows程序设计实验的课的关系，接触过vb[]但是没有专门去学他，因为习惯了c++里面的class[]int[]觉得vb的sub[]var 看着就不是很顺心。我是一个好奇心很强的人，突然看到了一个号称“.net是用于创建下一代应用程序的理想而又现实的开发工具”，而且主推c#语言，由于对c语言的一贯好感，我几乎是立刻对他产生了兴趣。我就开始了对c#的学习，任何语言都不是孤立存在的，所以数据交互是很重要的，暑假的时候我把我们这学期的课本数据库系统概论看了

一遍。我记得以前用c语言编程的时候，数据是在内存中申请空间，譬如使用数组等等。很耗费内存空间。这个时候就是数据库站出来的时候啦，于是我又装上了sql server[]以前学asp的时候用的是access[]那个时候只是照着人家做，理论是什么也不是很清楚。

通过一个暑假的学习，基本搞清楚了理论方面的东西，具体怎么用也不是很清楚。但是这为这学期的课程设计打下了铺垫。

来到学校后，随着这学期的数据库课程大赛开始了，我有一个看法就是我自己应该具备的能力不是我会多少，而是我应该具备快速学会东西的能力。遇到什么就学什么。我们有时候很容易被一些专业名词说吓着，包括什么建模，软件工程，数据分析，数据挖掘等等。我身边就有很多同学被这些纸老虎所唬住，而没有勇气去接触他们，总是说这个太难了之类的退堂鼓的话，他们低估了自己的潜力同时也压抑住了他们自己的好奇心。其实都是纸老虎，又不是什么国家科研难题，只是去用一些工具，发明工具是很难，但是用一个工具就容易多了[]just do it[]我记得我做这个数据库之前，我们老师说要做好前期分析，我就在网上搜索用什么分析工具好。最后我选择了roseuml建模工具。在此之前，我脑袋里面没有软件建模的思想，什么uml建模对我而言就是一张空白的纸。但是真正接触后并没有想象的那么难，有什么不懂的上网去搜索，这是一个信息横流的世界，有google[]baidu就没有不能解决的知识难题。以及后来的数据库分析的时候用到的powerdesigner也是一样。

开发的时候我想过用什么架构[]c/s模式？模式有很多，怎么选？我就上网搜索现在最流行的架构是什么。结果搜到了mvc架构，就是你啦。我决定用这个架构，不会，没关系，咱学[]just do it[]前期工作准备好后，加以实践。这个时候我更加深入的了解了利用操纵数据库的知识。并且对数据库里

面的存储过程有了比较深入的了解。经过大概2个多星期的奋斗，数据集的图书馆管理系统。并最后非常荣幸的获得了大赛的一等奖以及以及新技术应用奖。

与其临渊羡鱼，不如退而结网。这次数据库课程设计给我的最大的印象就是如果自己有了兴趣，就动手去做，困难在你的勇气和毅力下是抬不了头的。从做这个数据库开始无论遇到什么困难，我都没有一丝的放弃的念头。出于对知识的渴望，出于对新技术的好奇，出于对一切未知的求知。我完成了这次数据库课程设计，不过这只是我学习路上的驿站，的核心技术就是xml[至少微软是这么宣传的]，我会继续学习它，包括jave公司的j2ee我也很想试试，语言本来就是相通的[just do it]语言并不重要毕竟它仅仅是工具，用好一个工具并不是一件值得为外人道的事情，主要是了解学习思想。古语说的好：学无止境啊！

我很庆幸我参加了这次数据库大赛，让我确实打开了眼界。

（最后，很感激学校给了我们这次动手实践的机会，让我们学生有了一个共同学习，增长见识，开拓视野的机会。也感谢老师们对我们无私忘我的指导，我会以这次课程设计大赛作为对自己的激励，继续学习。毕竟学习就是一个just do it的过程！）

## 大数据课程总结篇七

一周的课程设计结束了，在这次的课程设计中不仅检验了我所学习的知识，也培养了我如何去把握一件事情，如何去做一件事情，又如何完成一件事情的方法和技巧。在设计过程中，和同学们相互探讨，相互学习，相互监督。我学会了运筹帷幄，学会了宽容，学会了理解，也学会了做人与处世，这次课程设计对我来说受益良多。

课程设计是我们专业课程知识综合应用的实践训练，这是我们迈向职业工作前一个必不可少的过程。“千里之行始于足下”，通过这次课程设计，我深深体会到这句千古名言的真正含义。我今天认真的进行课程设计，学会脚踏实地迈开这一步，就是为明天能稳健地在社会大潮中奔跑打下坚实的基础。我这次设计的科目是数据结。

数据结构，是一门研究非数值计算的程序设计问题中计算机的操作对象（数据元素）以及它们之间的关系和运算等的学科，而且确保经过这些运算后所得到的新结构仍然是原来的结构类型。“数据结构”在计算机科学中是一门综合性的专业基础课。数据结构是介于数学、计算机硬件和计算机软件三者之间的一门核心课程。数据结构这一门课的内容不仅是一般程序设计（特别是非数值性程序设计）的基础，而且是设计和实现编译程序、操作系统、数据库系统及其他系统程序的重要基础。通过这次模具设计，我在多方面都有所提高。

在界面设置中使用函数调用while其中文本显示颜色和背景颜色都可以任意按照自己的喜好，任意改变，但改变的时候必须采用标准英文大写，同时在制作显示菜单的窗口，大小根据菜单条数设计。最后采用printf输出程序设计界面。

这次的程序软件基本上运行成功，可以简单的建立链式循环链表，并进行输出，及循环语句的运用和选择语句的控制。由于时间和知识上的限制，使得程序规模相对较小，即功能还不很全面，应用也不很普遍。原来c语言可是涉及很多知识，而不是枯燥无聊的简单的代码部分而已，利用c语言方面的知识，我们可以设计出更完善的软件。

通过这次的课程设计，更是让我深刻认识到自己在学习中的不足，同时也找到了克服这些不足的方法，这也是一笔很大的资源。在以后的时间中，我们应该利用更多的时间去上机实验，加强自学的的能力，多编写程序，相信不久后我们的编程能力都会有很大的提高能设计出更多的更有创新的作品。

## 大数据课程总结篇八

这学期开始两周时间是我们自己选题上机的时间，这学期开始两周时间是我们自己选题上机的时间，虽然上机时间只有短短两个星期但从中确实学到了不少知识。上机时间只有短短两个星期但从中确实学到了不少知识。

数据结构可以说是计算机里一门基础课程，据结构可以说是计算机里一门基础课程，但我觉得我们一低计算机里一门基础课程定要把基础学扎实，定要把基础学扎实，然而这次短短的上机帮我又重新巩固了c语言知识，让我的水平又一部的提高。数据结构这是一门语言知识让我的水平又一部的提高。数据结构这是一门知识，纯属于设计的科目，它需用把理论变为上机调试。

纯属于设计的科目，它需用把理论变为上机调试。它对我们来说具有一定的难度。它是其它编程语言的一门基本学科。来说具有一定的难度。它是其它编程语言的'一门基本学科。我选的上机题目是交叉合并两个链表，对这个题目，我选的上机题目是交叉合并两个链表，对这个题目，我觉得很基础。刚开始调试代码的时候有时就是一个很小的错觉得很基础。刚开始调试代码的时候有时就是一个很小的错调试代码的时候误，导致整个程序不能运行，然而开始的我还没从暑假的状态导致整个程序不能运行，态转到学习上，每当程序错误时我都非常焦躁，态转到学习上，每当程序错误时我都非常焦躁，甚至想到了放弃，但我最终找到了状态，一步一步慢慢来，放弃，但我最终找到了状态，一步一学无止境，得自己懂得的知识很是不足，学无止境，以后还会更加的努力深入的学习。力深入的学习。

## 大数据课程总结篇九

两个星期的时间非常快就过去了，这两个星期不敢说自己有

多大的进步，获得了多少知识，但起码是了解了项目开发的部分过程。虽说上过数据库上过管理信息系统等相关的课程，但是没有亲身经历过相关的设计工作细节。这次实习证实提供了一个很好的机会。

通过这次课程设计发现这其中需要的很多知识我们没有接触过，去图书馆查资料的时候发现我们前边所学到的仅仅是皮毛，还有很多需要我们掌握的东西我们根本不知道。同时也发现有很多已经学过的东西我们没有理解到位，不能灵活运用于实际，不能很好的用来解决问题，这就需要我们不断的大量的实践，通过不断的自学，不断地发现问题，思考问题，进而解决问题。在这个过程中我们将深刻理解所学知识，同时也可以学到不少很实用的东西。

从各种文档的阅读到开始的需求分析、概念结构设计、逻辑结构设计、物理结构设计。亲身体验了一回系统的设计开发过程。很多东西书上写的很清楚，貌似看着也很简单，思路非常清晰。但真正需要自己想办法去设计一个系统的时候才发现其中的难度。经常做到后面突然就发现自己一开始的设计有问题，然后又回去翻工，在各种反复中不断完善自己的想法。

我想有这样的问題不止我一个，事后想想是一开始着手做的时候下手过于轻快，或者说是根本不了解自己要做的这个系统是给谁用的。因为没有事先做过仔细的用户调查，不知道整个业务的流程，也不知道用户需要什么功能就忙着开发，这是作为设计开发人员需要特别警惕避免的，不然会给后来的工作带来很大的麻烦，甚至可能会需要全盘推倒重来。所以以后的课程设计要特别注意这一块的设计。

按照要求，我们做的是机票预订系统。说实话，我对这个是一无所知的，没有订过机票，也不知道航空公司是怎么一个流程。盲目开始设计的下场我已经尝过了，结果就是出来一个四不像的设计方案，没有什么实际用处。没有前期的调查，



仅从指导书上那几条要求着手是不够的。

在需求分析过程中，我们通过上网查资料，去图书馆查阅相关资料，结合我们的生活经验，根据可行性研究的结果和客户的要求，分析现有情况及问题，采用client/server结构，将机票预定系统划分为两个子系统：客户端子系统，服务器端子系统。在两周的时间里，不断地对程序及各模块进行修改、编译、调试、运行，其间遇到很多问题：由于忘记了一些java语言的规范使得在调试过程中一些错误没有发现，通过这次课程设计，我对调试掌握得更加熟练了，意识到了程序语言的规范性以及我们在编程时要有严谨的态度，同时在写程序时如有一定量的注释，既增加了程序的可读性，也可以使自己在读程序时更容易。

我们学习并应用了sql语言，对数据库的创建、修改、删除方法有了一定的了解，通过导入表和删除表、更改表学会了对于表的一些操作，为了建立一个关系数据库信息管理系统，必须得经过系统调研、需求分析、概念设计、逻辑设计、物理设计、系统调试、维护以及系统评价的一般过程，为毕业设计打下基础。

很多事情不是想象中的那么简单的，它涉及到的各种实体、属性、数据流程、数据处理等等。很多时候感觉后面的设计根本无法继续，感觉像是被前面做的各种图限制了。在做关系模型转换的时候碰到有些实体即可以认为是实体又可以作为属性，为了避免冗余，尽量按照属性处理了。

物理结构设计基本没有碰到问题，这一块和安全性、完整性不觉就会在物理结构设计中添加一些安全设置：主键约束、check约束、default定义等。最后才做索引的部分，对一些比较经常使用搜索的列，外键上建立索引，这样可以明显加快检索的速度，最后别忘记重要的安全性设置，限制用户访问权限，新建用户并和数据库用户做相应的映射。

不管做什么，我们都要相信自己，不能畏惧，不能怕遇到困难，什么都需要去尝试，有些你开始认为很难的事在你尝试之后你可能会发现原来她并没有你以前觉得的那样，自己也是可以的。如果没有自信，没有目标，没有信心就不可能把事情做好，当其他人都在迷茫的时候，自己一定要坚信目标，大学毕业出去即面临找工作，从学习这个专业，到以后从事这方面的工作都需要不断地去学习去实践，这次实践可以给我们敲一个警钟，我们面临毕业，面临择业，需要这些实践经验，在困难面前要勇于尝试，这是这次课程设计给我的最大感想！

以上基本是这次实习的体会了，设计进行的非常艰难，编码非常不容易，才发现做一个项目最重要的不在于如何实现，而是实现之前的需求分析和模块设计。创新很难，有些流行的系统其实现并不难，难的在于对市场的分析和准确定位。设计，是一个任重道远的过程。

数据库课程设计大赛的尘嚣渐渐远去，怀着对这次大赛的些许不舍，怀着对当初课程设计开始时候的豪情万丈的决心的留恋，怀着通过这次课程设计积累的信心与斗志，我开始写这篇文章，为自己的足迹留下哪怕是微不足道但是对自己弥足珍贵的痕迹并期望与大家共勉。

首先，让我的记忆追溯到大二暑假，在老大的指引下()，我接触到microsoft产品。那个时候我已经学过vc和asp[]因为windows程序设计实验的课的关系，接触过vb[]但是没有专门去学他，因为习惯了c++里面的class[]int[]觉得vb的sub[]var看着就不是很顺心。我是一个好奇心很强的人，突然看到了一个号称“.net是用于创建下一代应用程序的理想而又现实的开发工具”，而且主推c#语言，由于对c语言的一贯好感，我几乎是立刻对他产生了兴趣。我就开始了对c#的学习，任何语言都不是孤立存在的，所以数据交互是很重要的，暑假的时候我把我们这学期的课本数据库系统概论看了

一遍。我记得以前用c语言编程的时候，数据是在内存中申请空间，譬如使用数组等等。很耗费内存空间。这个时候就是数据库站出来的时候啦，于是我又装上了sql server2019[]以前学asp的时候用的是access[]那个时候只是照着人家做，理论是什么也不是很清楚。

通过一个暑假的学习，基本搞清楚了理论方面的东西，具体怎么用也不是很清楚。但是这为这学期的课程设计打下了铺垫。

来到学校后，随着这学期的数据库课程大赛开始了，我有一个看法就是我自己应该具备的能力不是我会多少，而是我应该具备快速学会东西的能力。遇到什么就学什么。我们有时候很容易被一些专业名词说吓着，包括什么建模，软件工程，数据分析，数据挖掘等等。我身边就有很多同学被这些纸老虎所唬住，而没有勇气去接触他们，总是说这个太难了之类的退堂鼓的话，他们低估了自己的潜力同时也压抑住了他们自己的好奇心。其实都是纸老虎，又不是什么国家科研难题，只是去用一些工具，发明工具是很难，但是用一个工具就容易多了[]just do it!我记得我做这个数据库之前，我们老师说要做好前期分析，我就在网上搜索用什么分析工具好。最后我选择了roseuml建模工具。在此之前，我脑袋里面没有软件建模的思想，什么uml建模对我而言就是一张空白的纸。但是真正接触后并没有想象的那么难，有什么不懂的上网去搜索，这是一个信息横流的世界，有google[]baidu就没有不能解决的知识难题。以及后来的数据库分析的时候用到的powerdesigner也是一样。

开发的时候我想过用什么架构[]c/s模式?模式有很多，怎么选?我就上网搜索现在最流行的架构是什么。结果搜到了mvc架构，就是你啦。我决定用这个架构，不会，没关系，咱学[]just do it!前期工作准备好后，加以实践。这个时候我更加深入的了解了利用操纵数据库的知识。并且对数据库里面

的存储过程有了比较深入的了解。经过大概2个多星期的奋斗，数据集的图书馆管理系统。并最后非常荣幸的获得了大赛的一等奖以及以及新技术应用奖。

与其临渊羡鱼，不如退而结网。这次数据库课程设计给我的最大的印象就是如果自己有了兴趣，就动手去做，困难在你的勇气和毅力下是抬不了头的。从做这个数据库开始无论遇到什么困难，我都没有一丝的放弃的念头。出于对知识的渴望，出于对新技术的好奇，出于对一切未知的求知。我完成了这次数据库课程设计，不过这只是我学习路上的驿站，的核心技术就是xml[至少微软是这么宣传的]，我会继续学习它，包括jave公司的j2ee我也很想试试，语言本来就是相通的□just do it!语言并不重要毕竟它仅仅是工具，用好一个工具并不是一件值得为外人道的事情，主要是了解学习思想。古语说的好：学无止境啊！

我很庆幸我参加了这次数据库大赛，让我确实打开了眼界。

由于平时接触的都是一些私人项目，这些项目大都是一些类库，其他人的交流相对可以忽略不计，因此也就不考虑规范化的文档。实际上从学习的经历来看，我们接触的知识体系都是属于比较老或比较传统的，与现在发展迅速的it行业相比很多情况已不再适用，尤其是当开源模式逐渐走近开发者后更是如此。

虽然这次是一个数据库课程设计，由于本人在选择项目的时候是本着对自己有实际应用价值的角度考虑的，所以其中也涉及到一些数据库以外的设计。对于ooa/ood的开发模式有时不免要提出一些疑问□uml是设计阶段的工具，而它基本涵盖了软件设计的方方面面，也就是说按照这一软件工程的正常流程，在动手写第一句代码之前，开发人员已经非常熟悉软件产品了，这对于相当有经验的架构师一类人说可能会很容易，但是我们作为学生，连足够的编码经验都没有，却首先

被教授并要求先ooa再oop[]这样直接导致的问题就是文档与编码对不上号，在修改代码的时候基本不会再去审查文档和先前的分析。甚至根本就是现有代码再有文档，即便是这种情况，代码与文档还是不对应。不可否认，在传统软件工程的详细设计之前的项目过程中还是有很多利于项目开发的部分的。所以我就一直在寻找适合我——针对探究型项目——的开发模式，这次的项目也算是一次尝试，当然这个过程并不会太短。

回到数据库设计上了，这次的数据库设计我是严格按照数据库建模的步骤来进行的，老实说我并没有感觉这样的流程对开发带来多大的帮助，反倒是觉得将思维转化为图表很浪费时间。总体上来说这次的项目也不是很大，而且在数据库的设计上比较保守，也就是说实际上数据库设计还可以再完善完善的。随着我对计算机领域的拓宽和加深，我也会静下心来思考在接触计算机之前的行为，很多次我能深切感觉到，其实我的大脑(未于别人比较)本身就是在使用一种更接近关系数据库的方式来记忆，所以我很可恨自然的设计出符合三范式的表结构来，即便我不知道这些范式的确切含义。可能就像“范式不太容易用通俗易懂的方式解释”一样，在“让工具用图标表述我的思维”时费了一番力气。

从我作为项目的提出人和实现者来看，这是个失败的项目，结合几次教学项目的实践，发现这也已经不是第一次了。主观原因占多数，比如，尝试新的开发方式，根据设计花了太多的时间来抽象出公用的库而忽略业务逻辑。就这次项目而言，失败的原因有以下几点：

使用了新的开发环境(vim)[]这是首次在脱离高级ide的情况下编码。

使用了新的开发语言(python[]actionsript3)[]因为我一直比较喜欢“学以致用”，而且这样的“数据驱动型”软件的整套自实现的库都已经完成了，但是由于语言本身的差异，迁

移时问题很多，当发现这一点是，已没有多少有效剩余时间了。

编码流程的不妥，我比较喜欢从底层的库开始开发，因为一旦库测试通过，将很容易将它放到不同的表示层下。但如果库没有测试成功，将导致整个项目没有任何可视化模型，所以这次的项目无法提交“可运行的代码”。

实践目的的不同，我轻易不放弃锻炼的机会，事实上，有机会就一定要比以前有所突破，总是照搬以前的做法还不如就不做呢。这个前提是因为现在能完全用来的学习的时间比较多，等到工作时再这样做的可能性就很小了，因此当然要抓紧机会了。不过还有一个隐藏原因，总以为自己很了不起，其实“遇到的问题数跟人的能力是成正比的”。