

C语言实验实验报告(优秀5篇)

在当下社会，接触并使用报告的人越来越多，不同的报告内容同样也是不同的。大家想知道怎么样才能写一篇比较优质的报告吗？下面是小编给大家带来的报告的范文模板，希望能够帮到你哟！

C语言实验实验报告篇一

为什么那么多人学习C语言都半途而废呢？原因就是他们找不到正确的学习方法！在学习的过程中四处碰壁，兴趣和自信心逐渐被消耗殆尽。对他们来说学习C语言是一件很痛苦的事！小编整理了C语言实验一实验报告1000字范文，希望能帮助到您。

1、首先结合老师上课所讲内容，将课本知识点回顾一遍，并将重点知识点摘录到笔记本中，经常翻看，形成自己的知识结构体系，把握最基本的概念。

2、将老师评时布置的作业进行统一的整理，有些有代表性的程序代码可以摘录到笔记本上，一方面可以加深记忆，另一方面也方便自己后来的上机操作，可以经常翻看。

3、将书本上的例题，典型算法，以及老师布置的作业上面的程序全部自己敲一遍，有的可能需要2~3遍，做到最程序了然于心。并且时常复习，每天可利用各种空隙时间在手机上编程，也可规定一段时间专门用来编程，临近考试的时间，可以每天都编写一段时间，锻炼自己的手感，减少错误率。

4、最后根据老师的建议，就是做题，事实证明这个是极为正确的。通过做大量的上机实验题，发现自己的问题，如果是概念问题，那就继续回到课本，再将知识点巩固一遍，如果

是新的题型，那就记住，下一次便不会再犯错。通过做习题，你对知识的理解就会更加深一个层次，而且慢慢的你会发现，你开始有了c语言的思维，并且越来越喜欢上它，有时候分析出一个程序题，也是一种快乐。

最后，我想说，做题会使自己有质的飞跃，千万不能忽视做题的过程，还有就是一旦有不懂，一定要去问老师，老师常常不仅能帮你解决问题，还可以打开你的思维，做到举一反三。

对于c语言，可以说，我对它是“爱恨交加”。

第一个阶段，我对它是完全不来电的，可以说从来没有一门课让我如此抓狂，刚开始的时候看书都很吃力，而且很难看进去。整个人完全是懵圈的状态。

第二阶段，我开始对c语言有了一丁点兴趣，在期中考试过后，我意识到问题的严重性，如果自己再不想办法，就真的要出问题了。

第三阶段，我开始努力整理知识点，强迫自己去编代码，慢慢的我发现c语言跟我喜欢数学也是相通的，它们都极具思维性，有时候你看到自己编的程序运行成功，你会收获莫名的快感。

第四阶段，在期末模拟测验中，虽然我的成绩有了明显的提升，但仍未达到我的预期，这个时候，我便去向老师求救，老师建议我做题并给我了一本习题集，也正是这本习题集让我的成绩有了质的飞跃，也给了我继续前进的信心。

通过做题，我发现了很多问题，也遇见了很多新的题型。其实刚开始做的时候，会错很多，其实也挺失落的，但转念一想，发现错误是好事，以后便不会错了。慢慢的，我的正确率有了明显的提升。征服它的欲望也就越来越强烈，我时常告诉自己其它门课自己可以很轻松的学好，为什么让c语言成

为自己的一大败笔。

在这样的信念的支撑下，我熬过了期末复习的那段日子，说实话，那段时间整个人都是紧绷的状态，在进考场时，腿都发软。后来期末考试没有辜负我，总算对的起这么长时间的努力。

其实回头想想，也并没有觉得有多苦，我觉得最关键的还是要有信心，并且学会坚持。在期末后期，自己有点坚持不下去的感觉，但在和老师交流过后，又重拾了信心，记得老师说要想成功，你就必须比别人更加努力，后来自己一想，的确，如果不努力，你凭什么成功。最后我还得感谢老师，给了我前进的勇气，让我最终战胜了c语言。天道酬勤，经过这门课的学习，也坚定了我的信心，在未来的学习道路上，无论遇到什么困难，只要自己不放弃，就一定会收获意外的喜悦。

这个星期，我们迎来了c语言实训——一次至为重要的实训。在这个星期里，同学们都很认真的做着那20多道实训题目，遇到不懂的互相请教，或请问老师。非常感谢老师对我们的帮助，这次实训使我受益匪浅，收获了很多。通过c语言实训，让我加深了对c语言的了解，而不只是单单的在课本中学到的那些理论，平时乏味的课程，通过自己动手亲自编写，变的生动有趣，而在自己动手的过程中，出现的问题很多，比理论要难的多，当一个程序写完以后，经常会有很多错误而没法解决。不过，通过几天的实训，逐渐积攒了一些经验，有些错误可以很快就看出来。

这次实训有很大的收获，让我对c语言有了更深的认识，平时在课堂上学到的东西可以自己动手编写，将其转化成一些实用的技能。如果是一个程序写完一处错误也没有，会有种成就感，于是兴趣就来了，兴趣来了，自然学的东西也就多了，能把理论变成实际的技能，让我对c语言有了浓厚的兴趣和更

深层的认识□ c语言是一个有序的学习，学了最基本的. 替换，然后扩展到循环，嵌套，条理很清楚，不是一个零散的知识，实际上所有的课程都如此，不过通过实训我也知道了自己的不足，存在的很多问题。比如自己写的写的小程序出了问题，不会解决了就叫老师帮忙，虽然说一定意义上增加了师生之间的感情，但是会养成一种依赖的心理，碰到问题了第一个想到的是求助而不是自己独立解决，所以以后要多多锻炼自己的信心和增加自己的能力，争取做到老师不在身边的时候也可以完成一些简单的程序编写与错误排除。还有自己的基础知识不扎实，遇到的问题，没有很好的逻辑思维，亲自编写一个陌生的程序的时候会有种无法下手的感觉，找不到突破口。通过实训，逐渐理清了顺序，对于简单的程序和一些相对比较繁琐的嵌套，循环，不在是看着一头雾水。其实只要理清了思路，把基础知识掌握了，然后有条不紊的分析，一步一步理解□c语言还是很有意思的课程。自己亲自动手编写程序让我增加了对c语言程序开发环境的了解，在上课的时候老师就讲，学习c语言最重要的是学习c语言的逻辑思维，不管以后从事什么行业，学习c语言都对自己的职业很有帮助，如果是从事编程工程工作的话。当初在初步接触c语言的时候，看着一堆“奇形怪状”的符号，觉得甚是无聊，通过这次实训，摆脱了那种似懂非懂的状态!感谢学校安排这次实训和老师的耐心讲解，让我学到了很多知识，在实训过程中，同学之间的相互探讨，老师的循循善诱，最终让我们达到了举一反三的效果，在学知识的同时，也增加了同学老师之间的感情。希望以后还会有更多类似的实训课程，在有限的大学时间内学到更多的实用技能，为以后的工作打下一个良好的基础。

这次实训似乎是对我巨大的考验，程序在不慌不忙的进行着，按照我自身的进度，提前完成是没有问题的，但是很多客观条件让我不得不在最后的时间里才上交的作业。每当我把程序写完以后，却由于每一台电脑的识别格式不一样而导致我所写的文件一次次的被损坏，尽管这样，我仍然没有放弃，

最后还赶在截止时间到来之前上交了我的作业!

这一次编写大作业的程序又让我感受到学程序可以锻炼缜密的思维了。因为平时练习的都是小程序，所以句段比较少，一些错误都很容易被检查出来，但是这次实训却是上百段的语句，难以检查，似乎也没有经验去怎么检查，那一刹那感觉到很急，但又不知如何是好，可是又不轻易的向老师询问，因为好像蛮简单，相信自己能够检查出来，所以我一般都是独立思考的完成的，只有少部分是在老师和与同学讨论下完成的。在编写程序中，真的是细节决定成败，因为不管程序的语句是多是少，但是只要有一条语句或语法不正确，就不能使程序得到所需要的结果。没能够使自己编写出来的程序等到想要的结果，意味着这是一个不理想的程序，对于本次编写来说则是失败的。

最后，非常感谢学校给我们这次实训的机会。极高的锻炼了我们的逻辑思维能力。人非生而知之，虽然我们现在的知识结构还很差，但是我们知道要学的知识，一靠努力学习，二靠潜心实践。没有实践，学习就是无源之水，无本之木。在剩下的时间里，我们会更加努力的学习。

在这个星期里，我们在专业老师的带领下进行了c语言程序实践学习。在这之前，我们已经对c语言这门课程学习了一个学期，对其有了一定的了解，但是也仅仅是停留在了解的范围，对里面的好多东西还是很陌生，更多的在运用起来的时候还是感到很棘手，毕竟，万事开头难嘛。

由于时间的关系，我们的这次实践课程老师并没有给我们详细的介绍，只是给我们简单的介绍了几个比较重要的实际操作。上机实验是学习程序设计语言必不可少的实践环节，特别是c语言灵活、简洁，更需要通过编程的实践来真正掌握它。对于程序设计语言的学习目的，可以概括为学习语法规定、掌握程序设计方法、提高程序开发能力，这些都必须通过充分的实际上机操作才能完成。学习c程序设计语言除了课堂

讲授以外，必须保证有不少于课堂讲授学时的上机时间。因为学时所限，课程不能安排过多的统一上机实验，所以希望学生有效地利用课程上机实验的机会，尽快掌握用c语言开发程序的能力，为今后的继续学习打下一个良好的基础。个人认为课程上机实验的目的，不仅仅是验证教材和讲课的内容、检查自己所编的程序是否正确，课程安排的上机实验的目的可以概括为如下几个方面：

1、加深对课堂讲授内容的理解

课堂上要讲授许多关于c语言的语法规则，听起来十分枯燥无味，也不容易记住，死记硬背是不可取的。然而要使用c语言这个工具解决实际问题，又必须掌握它。通过多次上机练习，对于语法知识有了感性的认识，加深对它的理解，在理解的基础上就会自然而然地掌握c语言的语法规则。对于一些内容自己认为在课堂上听懂了，但上机实践中会发现原来理解的偏差，这是由于大部分学生是初次接触程序设计，缺乏程序设计的实践所致。

学习c语言不能停留在学习它的语法规则，而是利用学到的知识编写c语言程序，解决实际问题。即把c语言作为工具，描述解决实际问题的步骤，由计算机帮助我们解题。只有通过上机才能检验自己是否掌握c语言、自己编写的程序是否能够正确地解题。

通过上机实验来验证自己编制的程序是否正确，恐怕是大多数同学在完成老师作业时的心态。但是在程序设计领域里这是一定要克服的传统的、错误的想法。因为在这种思想支配下，可能你会想办法去“掩盖”程序中的错误，而不是尽可能多地发现程序中存在的问题。自己编好程序上机调试运行时，可能有很多你想不到的情况发生，通过解决这些问题，可以逐步提高自己对c语言的理解和程序开发能力。

2、熟悉程序开发环境、学习计算机系统的操作方法

一个c语言程序从编辑、编译、连接到运行，都要在一定的外部操作环境下才能进行。所谓“环境”就是所用的计算机系统硬件、软件条件，只有学会使用这些环境，才能进行程序开发工作。通过上机实验，熟练地掌握c语言开发环境，为以后真正编写计算机程序解决实际问题打下基础。同时，在今后遇到其它开发环境时就会触类旁通，很快掌握新系统的使用。

3、学习上机调试程序

完成程序的编写，决不意味着万事大吉。你认为万无一失的程序，实际上机运行时可能不断出现麻烦。如编译程序检测出一大堆错误。有时程序本身不存在语法错误，也能够顺利运行，但是运行结果显然是错误的。开发环境所提供的编译系统无法发现这种程序逻辑错误，只能靠自己的上机经验分析判断错误所在。程序的调试是一个技巧性很强的工作，对于初学者来说，尽快掌握程序调试方法是非常重要的。有时候一个消耗你几个小时时间的小小错误，调试高手一眼就看出错误所在。

通过这次为数不多的几天计算机实践学习，我们了解了一些关于c语言的知识，理解巩固了我们c语言的理论知识，着对我们将来社会工作将会有莫大的帮助。同时它让我知道，只要你努力，任何东西都不会太难。

回首这学期的c语言学期经历，其实还是比较感慨的，从最开始的懵懂无知到最后机试的满分，我自己也觉得有点不可思议。

刚开始接触这门课的时候，大概我和许多同学一样都很不喜欢这种枯燥的程序学科，感觉也是比较无奈的。说一个比较功利性的原因，无非就是拿学分，想考高分升绩点，所以即使是不喜欢，也要努力学好这门课。

我觉得很多同学说难无非就是程序练得少了，其实只要书本的基础性知识记住了，程序就纯粹是靠多练了。有好多同学常常在上课的时候睡觉，这个其实很不好，可能他们觉得下课自己看书也能学会，但那无疑不全面，效率也不高。我上课一般都算比较认真，需要听的都会特别关注，有时候需要用手机拍一下一些特殊或者总结行的关键知识以及一些典型的程序，这样下课再仔细琢磨，再自己练习，效果无疑会更好。

还有就是C语言布置的课后练习上机作业特别重要，我每次都会找空闲时间来敲程序，写作业，而且必须要独立完成，我知道有很多同学都复制过别人的作业，但我一直坚持独立完成，有时候想一道程序的填空想半天，实在是弄不明白的话就请教别的同学，反正是得弄懂。作业是一定要做的，而且要自己做，坚持练习。

有很多时候也会自己一个人看书，然后敲一些程序，别人问我的程序也会敲，所以除了作业，我其实也敲了很多别的练习程序。没结课的时候，有好些同学都能经常见到我在电脑前敲程序，这就是练习的重要性，练得多了，程序敲起来就更顺手，考试的时候速度其实很重要，如果你敲得快，你完全有时间检测选择和填空的程序题。我考试的时候就是速度很快，写完后，还有将近半个多小时，然后我就开始敲程序检测选择填空，改了好几道，所以说熟练度的问题决定你的效率。

另外，考前复习也是比较重要的，可能你结课后基本没怎么做过C语言题，所以好些会忘记，这时候，要先翻一遍书，把基础知识再复习一遍，这很重要，尤其是一些概念，以免做题时忘记，敲程序也需要这些基础性的东西。找一些选择题来看，最好是那些概念性的东西，因为这些是需要背的，别的都可以计算。当然，复习还是要敲程序的，即使你觉得不需要再看了，也要敲几个熟悉一下，找找感觉，不会太久没敲显得突然，不然会忘步骤，而且你还不知道忘的是哪部分。

我的复习步骤就是这几步，学习经验其实也比较简单，如果大家努力其实都可以做到。C语言学习虽然比较枯燥，但这并不能成为你学不好的理由，而且，只要你学好了，你也就不会觉得乏味无趣了。

C语言实验实验报告篇二

二十一世纪是计算机的世纪，是一个全新的世纪，计算机正逐步入侵我们的生活，刚开始接触计算机的时候感觉很新奇，那时觉得计算机很神秘，于是就产生了以后一定要熟练掌握计算机的念头。怀着这个愿望，我选择了计算机科学与技术为专业，有点兴奋又有点担忧，兴奋的是以后可以经常接触和了解计算机了，担忧的是我几乎没有任何基础，怕跟不上大家的步伐。有死党说我杞人忧天，结果证实我“忧”得不无道理。依然清楚地记得第一学期的上机课，每次上机老师都计分，所以经常手忙脚乱，用“惨不忍睹”来形容是最合适不过了。

紧张的一个学期过后，我们迎来了C语言的学习和一位风趣博学的老师。虽然刚开始学习C语言的时候大家都一筹莫展，但在老师的鼓励和教育下一步步地往前迈，很令人欣慰。在这之后的学习中仍然困难重重，然而“坚持到底就是胜利”这个坚定的信念激励着我们勇往直前。C语言课本的学习总算结束了，多么崎岖的一条学习之路。

完成了计算器的图形很有成就感，毕竟六月天坐在机子前不是令人愉快的事。接下来要进行计算器内部运算程序的设计，说实话，我根本就不知道要做，更不清楚如何着手，不是我妄自菲薄，独立完成这个程序设计根本就是不可能的事，可想而知只能寻求帮助了，但这个程序不是轻而易举就能完成的。有些积极的同学从网上搜索了好多有关这方面的资料，拼拼凑凑，进行了多次讨论，终于完成了这部分的设计，在

同学的互相帮助下，大多数人都完成了最重要的一部分程序设计。

接下来就是用循环语句使按键上的数字有闪烁的功能，这个因为之前老师有过提点，所以相对简单一些。

完成了整个程序设计，这次课程设计就接近尾声了，不过最后的课程报告着实让我们苦恼了一两天。小小的抱怨一下应该没关系吧。

整个课程设计终于完成了，完成的好辛苦，好艰难，但还是顺利的完成了。我们付出了很多心血和汗水，但我认为我们得到的更多。如果没有同学们的互相帮助，这次课程设计就不会这么顺利，这种同舟共济的团队精神深深震撼了我，

《礼记学记》上也有讲过：“独学而无友，则孤漏而寡闻”。同时有些同学积极的态度让我感到一丝惭愧，他们所体现出的坚持不懈的品质更令我折服。当然这次程序设计也让我看到了自己基础的薄弱。古人有诗云：“路漫漫其修远兮，吾将上下而求索”。从今天起我将不断继续求索，学习之路也好，人生之路也罢。

总之我受益匪浅。

c语言实验实验报告篇三

指导教师：_____ 成绩：_____

实验四 函数

一、 实验目的

1、掌握函数定义、调用和声明的方法

2、掌握实参和形参之间的传递方式

3、函数的嵌套调用

二、 实验内容

1、 写一个函数，将两个字符串连接。（习题8.6）

2、 编写一个函数，由实参传来一个字符串，统计此字符串中字母、数字、空格和其他字符的个数，在主函数中输入字符串以及输出上述的结果。（习题8.9）

3、 请将实验三中的实验内容三改正后，再改写成函数形式（排序部分）。

三、 实验步骤与过程

四、 程序调试记录

c语言实验实验报告篇四

指针是c语言中广泛使用的一种数据类型，运用指针编程是c语言最主要的风格之一。利用指针变量可以表示各种数据结构；能很方便地使用数组和字符串；并能象汇编语言一样处理内存地址，从而编出精练而高效的程序。指针极大地丰富了c语言的功能。学习指针是学习c语言中最重要的一环，能否正确理解和使用指针是我们是否掌握c语言的一个标志。同时，指针也是c语言中最为困难的一部分，在学习除了要正确理解基本概念，还必须要多编程，上机调试。只要作到这些，指针也是不难掌握的。

指针的基本概念在计算机中，所有的数据都是存放在存储器中的。一般把存储器中的一个字节称为一个内存单元，不同

的数据类型所占用的内存单元数不等，如整型量占2个单元，字符型量占1个单元等，在第二章中已有详细的介绍。为了正确地访问这些内存单元，必须为每个内存单元编上号。根据一个内存单元的编号即可准确地找到该内存单元。内存单元的编号也叫做地址。既然根据内存单元的编号或地址就可以找到所需的内存单元，所以通常也把这个地址称为指针。内存单元的指针和内存单元的内容是两个不同的概念。可以用一个通俗的例子来说明它们之间的关系。我们到银行去存取款时，银行工作人员将根据我们的帐号去找我们的存款单，找到之后在存单上写入存款、取款的金额。在这里，帐号就是存单的指针，存款数是存单的内容。对于一个内存单元来说，单元的地址即为指针，其中存放的数据才是该单元的内容。在c语言中，允许用一个变量来存放指针，这种变量称为指针变量。因此，一个指针变量的值就是某个内存单元的地址或称为某内存单元的指针。图中，设有字符变量c其内容为“k”(ascii码为十进制数75)其c占用了011a号单元(地址用十六进制数表示)。设有指针变量p其内容为011a这种情况我们称为p指向变量c或说p是指向变量c的指针。严格地说，一个指针是一个地址，是一个常量。而一个指针变量却可以被赋予不同的指针值，是变。但在常把指针变量简称为指针。为了避免混淆，我们中约定：“指针”是指地址，是常量，“指针变量”是指取值为地址的变量。定义指针的目的是为了通过指针去访问内存单元。

一个数组或一个函数的首地址有何意义呢？因为数组或函数都是连续存放的。通过访问指针变量取得了数组或函数的首地址，也就找到了该数组或函数。这样一来，凡是出现数组，函数的地方都可以用一个指针变量来表示，只要该指针变量中赋予数组或函数的首地址即可。这样做，将会使程序的概念十分清楚，程序本身也精练，高效。在c语言中，一种数据类型或数据结构往往都占有一组连续的内存单元。用“地址”这个概念并不能很好地描述一种数据类型或数据结构，而“指针”虽然实际上也是一个地址，但它却是一个数据结构

的首地址，它是“指向”一个数据结构的，因而概念更为清楚，表示更为明确。这也是引入“指针”概念的一个重要原因。

指针变量的类型说明

对指针变量的类型说明包括三个内容：

- (1) 指针类型说明，即定义变量为一个指针变量；
- (2) 指针变量名；
- (3) 变量值(指针)所指向的变量的数据类型，

其一般形式为：类型说明符*变量名；

其中，*表示这是一个指针变量，变量名即为定义的指针变量名，类型说明符表示本指针变量所指向的变量的数据类型。

例如 `int*p1;`表示 `p1` 是一个指针变量，它的值是某个整型变量的地址。或者说 `p1` 指向一个整型变量。至于 `p1` 究竟指向哪一个整型变量，应由向 `p1` 赋予的地址来决定。

再如：

`static int*p2; /*p2是指向静态整型变量的指针变量*/`

`float*p3; /*p3是指向浮点变量的指针变量*/`

`char*p4; /*p4是指向字符变量的指针变量*/` 应该注意的是，一个指针变量只能指向同类型的变量，如 `p3` 只能指向浮点变量，不能时而指向一个浮点变量，时而又指向一个字符变量。

指针变量的赋值

指针变量同普通变量一样，使用之前不仅要定义说明，而且必须赋予具体的值。未经赋值的指针变量不能使用，否则将造成系统混乱，甚至死机。指针变量的赋值只能赋予地址，决不能赋予任何其它数据，否则将引起错误。在c语言中，变量的地址是由编译系统分配的，对用户完全透明，用户不知道变量的具体地址[]c语言中提供了地址运算符&来表示变量的地址。其一般形式为：&变量名；如&a表示变量a的地址[]&b表示变量b的地址。变量本身必须预先说明。设有指向整型变量的指针变量p[]如要把整型变量a的地址赋予p可以有以下两种方式：

(1) 指针变量初始化的方法inta;

```
int*p=&a
```

(2) 赋值语句的方法inta;

```
int*p;
```

```
p=&a
```

指针变量的运算

指针变量可以进行某些运算，但其运算的种类是有限的。它只能进行赋值运算和部分算术运算及关系运算。

1. 指针运算符

(1) 取地址运算符&

取地址运算符&是单目运算符，其结合性为自右至左，其功能是取变量的地址。在scanf函数及前面介绍指针变量赋值中，我们已经了解并使用了&运算符。

(2) 取内容运算符*

取内容运算符*是单目运算符，其结合性为自右至左，用来表示指针变量所指的变量。在*运算符之后跟的变量必须是指针变量。需要注意的是指针运算符*和指针变量说明中的指针说明符*不是一回事。在指针变量说明中，“*”是类型说明符，表示其后的变量是指针类型。而表达式中出现的“*”则是一个运算符用以表示指针变量所指的变量。

```
main{
```

```
inta=5,*p=&a
```

```
printf("%d",*p);
```

c语言实验实验报告篇五

一周的c语言实训，虽然时间略显仓促，但却让我学到了很多实际运用的技能！

一、心得与感想

首先，在实训中的第一步是编写功能较为单一的小程序，虽然是小程序，但所用的知识却都是我们本学期学习的重点知识。在做题的过程中让我们把这些知识复习了一遍，强化了知识！

其次，这次作中所碰到的题目，在实训中已经接触过，所以我们都比较深知这些题目的要求和算法，因次在编写小程序时比较快，而且算法也是比较简化，运算准确。鉴次我明白很多编程题要经过多次编写修改与调试，才会得到最简略的算法。

再次，平时我们学c语言视乎都是为了考试或者做一些比较常

规的题目，而这次实训却给我们的将所学知识运用于解决实际问题的机会，让人更有激情，这就是编写软件的雏形，也让我们感受到了编写软件程序的乐趣。但是不仅仅是如此，更为重要的是在运用所学知识的过程中，记住这些知识，并能够广泛的运用！

虽然我这次是编写一个菜单的程序，但是如果我么留心的话，我们身边有许多电子设备都是内置有程序的，我们也可以在此次专题实训过后，自行确定其他的专题进行编写程序，这样可以让我们的知识更加深化丰富！

这次实训似乎是对我巨大的考验，程序在不慌不忙的进行着，按照我自身的进度，提前完成是没有问题的，但是很多客观条件让我不得不在最后的时间里才上交的作业。每当我把程序写完以后，却由于每一台电脑的识别格式不一样而导致我所写的文件一次次的被损坏，尽管这样，我仍然没有放弃，最后还赶在截止时间到来之前上交了我的作业！

这一次编写大作业的程序又让我感受到学程序可以锻炼缜密的思维了。因为平时练习的都是小程序，所以句段比较少，一些错误都很容易被检查出来，但是这次实训却是上百段的语句，难以检查，似乎也没有经验去怎么检查，那一刹那感觉到很急，但又不知如何是好，可是又不轻易的向老师询问，因为好像蛮简单，相信自己能够检查出来，所以我一般都是独立思考的完成的，只有少部分是在老师和与同学讨论下完成的。

在编写程序中，真的是细节决定成败，因为不管程序的语句是多是少，但是只要有一条语句或语法不正确，就不能使程序得到所需要的结果。没能够使自己编写出来的程序等到想要的结果，意味着这是一个不理想的程序，对于本次编写来说则是失败的。