

# 2023年软件课程设计结课报告 课程设计 总结课程设计报告(精选5篇)

报告是指向上级机关汇报本单位、本部门、本地区工作情况、做法、经验以及问题的报告，那么，报告到底怎么写才合适呢？下面我就给大家讲一讲优秀的报告文章怎么写，我们一起来了解一下吧。

## 软件课程设计结课报告篇一

通过学习dsp课程我们都知道dsp实际上也是一种单片机，它同样是将中央处理单元、控制单元和外围设备集成到一块芯片上。但dsp有别于普通的单片机，它采用了多组总线技术实现并行运行机制，从而极大地提高了运算速度，也提供了非常灵活的指令系统。

此次课程设计是从我们做完单片机的课程设计后开始的。实际上在单片机的课设时，我们就着手准备设计资料。按设计要求，按部就班地进行设计，基本上用了一周的时间。每天我们都要在固定的教室碰面，安排一天的设计任务，每人都有明确的分工。而我的分工就是和其他的两个组员按照事先画好的电路图焊接电路板。这个工作看起来很容易，毕竟我们大一的时候就进行过类似的焊接电路板的电子工艺实习。所以我以为会很得心应手，但就是因为有那么点轻视的味道在里面，导致了我后面的功亏一篑，一周的成果不得不全盘重来。

在刚开始进行焊接的时候我们就犯了一个整体布局的错误，因为一开始以为会有主芯片焊在电路板上，所以我们的芯片的布局比较集中，这就直接导致了后来我们接线的困难，但为了避免更大的麻烦，我们还是继续了下去。当时完全没有考虑到一个基本常识，那就是像这种直接用飞线连接的情况下使用过于密集的连接方法不仅不容易焊接，而且电器性能

非常的差，极其容易导致短路和强烈的电子干扰。

就这样，我们在之后的一周时间里完成了板子的初步焊接工作，就在我们兴高采烈的拿去实验的时候，结果确是指示灯毫无反映，这下我们就迷糊了，什么原因呢？查吧！于是乎我们开始了一个下午的仔细的检查工作，还别说，我们小组特别认真的按照电路图来一一检查，还真被我们检查到了好几处明显的错误，特别是好几个电容没有接地，这种低级错误真是不应该犯啊。我们自己看到这个错误都觉得很不应该，不过知错能改，善莫大焉。我们很快的改正了这个错误。于是第二次拿去测试，想不到这次问题更糟，发光二极管直接一股青烟给烧了。我们当时这是傻了眼了，只好又回来检查。

检查后才发现我们电路图都画错了，原来我们把发光二极管直接接到12v电源上了，不烧才怪。经过这两次的错误，我们总结了一下我们出现的错误，大部分都是很低级的错误，这真是我们不认真造成的啊。于是我们又一次认真的检查了一下电路板，这下果然被我们发现了好几处短路的地方。但是由于一开始的整体排列就没有弄好，导致了现在都无法改动了。终于在我们一致讨论后，我们终于决定干脆推倒重来，把功率放大模块整体重新焊接过。这样虽然可能速度比较慢，但正确性很高，这其实也是无奈之举。要不是当初焊接的时候没焊好，导致问题多多，现在也不至于要重新来过，不过还好，我们在接受这次惨痛的教训后终于明白了一个道理，工作一定要踏踏实实的去做，特别是学我们这行的，一点马虎的也不能有，宁可重来，也不能交上一分马虎之中做出的东西！在这样的信念下，我们一直忙到晚上七点，终于是完成重新焊接的工作了，为了保险起见，我们又一次重新完整的检查了一遍系统，在最后确定没有问题后，我们终于欣慰的走出了教室的大门，来到了食堂吃饭，那一刻感觉食堂的饭怎么都比平常好吃多了，呵呵，这大概就是所谓的自我成就吧。

通过这次课程设计，我懂得了做一件事情一定要细心仔细，

要不然本来很简单的一件事情也会做的很糟糕，而且一定有怀疑的精神，我们这次二极管烧了就是因为太迷信自己设计的电路图了，没有注意到那个明显的设计失误，才导致了这种结果。我希望以后能有更多的课程设计来给我们锻炼，这样我们将来走向社会才会有更强的竞争力。我会更加的珍惜以后的这种学习的机会的！

篇三

## 软件课程设计结课报告篇二

学生成绩管理是高等学校教务管理的重要组成部分，其内容较多，要完成能够实现人员的信息录入、查询、修改、删除、统计、保存、显示等操作。系统具体实现的功能如下：

1. 添加新的联系人
2. 删除所有联系人
3. 删除某个联系人
4. 查询某个联系人
5. 查询所有联系人
6. 将电话本导出到文件
7. 从文件导入电话本
8. 查询并编辑某个联系人
9. 全班同学的平均分

## 大概的算法分析

### 学生成绩管理软件的数据结构

学生成绩一般都是以每个成员为单位进行管理的，一个学生学号用一个结点来表示。考虑到添加、查询和删除的方便，这些以学号为单位的结点用链表方式组成一个主目录。每个结点有四个域：

- (1) 学生代号；
- (2) 课程情况；
- (3) 成绩信息管理情况；

用代码实现输入及输出添加新的联系人的学号，姓名，性别，生日、语文成绩，数学成绩；完成添加工作。

#### (2) 删除某个联系人

首先输入要删除的姓名，然后要在系统中查看是否有此记录，再用分支结构实现删除链表中此员工结点并修改学校员工链表及未找到这条记录；因为可能系统中不存在此记录。

#### (3) 删除所有联系人

此功能要逐条删除联系人，则我们采用分支、循环结构实现首先遍历是否有给记录，输出读取记录或没有该记录；再到文件是否读完，若没读完则要循环继续读取，否则已删除了所有记录。

#### (4) 查询某个联系人

因此功能会出现还没有任何记录、未找到这条记、输出该联系人信息三种情况，我们则选用简单的三分支结构；首先要

请输入要查询联系人的姓名，输出有记录或还没有任何记录，有记录则遍历所有人，最后是未找到这条记录，输出该联系人信息。

### (7) 从电话本导入文件的流程图

首先查看是否有记录，输出没有任何记录或打开“”，历遍链表，、关闭文件，。

其余的函数功能实现相似则不重复。

### 心得体会

刚放假不久，我们组的积极分子们就张罗着要开始编写程序了，编写这个程序我们经历了讨论选题，编写代码，调试，写设计报告等。在这期间我们查阅了很多相关的资料，曾盯着课本发呆过，我们成员之间也曾进行过激烈的口舌论战。经过这个暑假来的实践学习，使我们对c++语言有了更进一步的认识和了解，要想学好它，则重在实践，要通过不断的上机操作及实例运用才能更好地深入体会它，当然通过实践，我们也发现我们的好多不足之处。就比如我们对c++语言的一些标准库函数不太了解，还有对函数调用的正确使用不够熟悉，一些经常出现的错误也不能很好地处理。我们编写这个程序使用的是比较简单的函数及结构，由于我们的水平有限，还不能很好的使用一些效率高的方式来编写，所以也没有突出的创新点。在做题的时候我们都是分工合作。分工完成结构体、函数设计、算法分析等，这些过程中总是有很多错误出现，有好多的小问题都是我们粗心大意造成的，所以设计程序一定要仔细，不容一点马虎。当然也有大问题，关于文件的操作，类的使用等等这些使用我们都琢磨了很久，在解决错误时我们有时几乎焦头烂额了。但是我们与队友之间都会不断的相互鼓励、支持，彼此都努力维护着将作业写完。虽然这设计有点枯燥无味，但是当最后我们大功告成时，回味起来还是美滋滋的。总的来说我们真的是在设计的过程

中受益匪浅。