

## Talking about C Language Programming--Taking Address Book Design as an Example

Jinpeng CHEN Zhengde BAO Chenxi Li

School of Computer and Software, Jincheng College, Sichuan University, Chengdu, Sichuan, 611731

### Abstract

As the basic language in computer field, C language is also a compulsory course for computer majors in major universities in China. How to design a complete big program is a big problem for students in Colleges and universities to solve. In this paper, by taking the specific address book design as an example, the basic ideas of C language programming are discussed. From a practical project, the needs of customers are analyzed in detail, then the planning and design are carried out according to the needs of customers. Finally, the system and function design are carried out according to the planning and design formulated by ourselves.

### Key Words

Programming, Functions, Customers

DOI:10.18686/jsjxt.v1i2.639

## 浅谈 C 语言程序设计——以通讯录设计为例

陈金鹏 鲍正德 李晨曦

四川大学锦城学院计算机与软件学院, 四川, 成都, 611731

### 摘要

C 语言作为计算机领域的基础语言,也是国内各大高校计算机专业的一门必修课,如何设计一个完整的大程序是各高校同学所要解决的一大难题。本文通过以具体的通讯录设计为案例,浅谈了 C 语言程序设计的基本思路,从一个实际项目出发,对客户的需求进行详细分析,然后根据客户需求进行规划设计,最后根据自己拟定的规划设计,进行系统、功能设计。

### 关键词

程序设计; 函数; 客户

### 1. 引言

C 语言作为计算机学习的一门基础语言,在计算机领域内,无论做什么都离不开 C 语言,现在高校的“C 语言程序设计”是属性必修课,尤其对非计算机专业的学生,开设 C 语言课程非常必要。<sup>[1]</sup>而当今本科阶段教育,对 C 语言的掌握程度要求并不算太高,很少有达到独立设计出一个小程序的要求。C 语言是一门实践性很强的课,因此在我们学习的时候,不需要死记一些语法,而应该多上机操作,完成一个个小程序。<sup>[2]</sup>在 C 语言的大程序设计当中,离不开小程序设计,可以说大程序是由很多个小程序所组成的,而这一个一个的小程序,

又是由一个一个小板块构成的。所以,从每一个小板块开始做好,才是我们大程序设计的良好开端,完整设计一个程序,从什么角度去思考程序设计,满足客户需求,达到用户想要的效果,是我们完成大程序设计要达到的目的。

### 2. 需求分析

大型的程序设计,往往是在于客户提出了需求,然后程序员需要按照客户提出的需求,做出合理的分析并提出满足这个需求的设计。例如,现在客户提出了设计一个通讯录系统,我们首先应该考虑通讯录有哪些内容,需要设计哪些基本的功能。现在的计算机领域上,软件

的产品越来越多,然而不同的软件却无法达到客户的各种需求。企业现在的发展目标,应该根据客户的需求来对软件进行开发,而不是盲目去的追求数量。<sup>[3]</sup>

## 2.1 功能需求分析

功能需求分析,是在一个项目中,必须要拥有的一部分,也是一个项目的主要的基本功能,一般会涉及到对核心内容数据的修改,需求分析一般先从核心和主要部分开始入手分析。开发一个软件项目,需求分析是核心点,软件的开发者需要对这个项目进行深入的了解,要清楚这个项目的限制是什么,怎么样让用户可以很好地去操作这个系统。<sup>[4]</sup>

一个简单的通讯录系统设计,其数据应当有姓名、电话号码以及电话号码类型等,其功能应该有增加、删除、修改、查找以及拨号功能,有了以上基本的数据类型以及功能,从这些功能的角度去设计一个通讯录,才能满足客户需求。然后,我们在根据不同的功能,单独分析每一个功能,做好每一个功能,将这些功能组合起来就是一个大程序。以通讯录设计为例,以下是我们进行客户需求分析后,得到的通讯录的基本功能:

### 2.1.1 增加数据的功能

程序能够录入一条新的数据(一个数据包括:姓名、电话号码、分类、电子邮件)。为了避免用户失误操作导致一个人的信息重复录入,占用通讯录的存储空间,因此我们设计功能时还需要进行去重复操作。由于系统空间存在上限,因此我们需要规定通讯录信息的录入上限。

### 2.1.2 查看数据的功能

用户想要通过一个人的姓名查找到想要找的人的信息。

### 2.1.3 修改的功能

对于录入不是完全正确的数据,或者是个人信息发生了改变,则需修改这个人的信息。

### 2.1.4 删除的功能

对于成就过时或更换号码的情况,对信息进行删除。

以上的几个功能,我们通过设计一个菜单进行整体

管理,当用户想使用某一功能时,可以通过选择其对应的编号进行操作,以上功能也是对通讯录的核心数据进行修改,也是一个通讯录最基础必要的功能设计。完成了客户功能需求分析,找到了客户所需要的东西,完成了对核心数据的控制,这是我们做程序设计的第一步。

## 2.2 非功能需求分析

非功能需求分析,则是一些不涉及到核心数据修改的功能,包括系统的性能,维护等。

在生活中,我们希望通过通讯录快速查找到一个人的电话号码,然后进行拨号操作,因此通讯录系统设计中,我们单独设计了一个拨号的功能:我们在使用手机时,希望通过找到一个人的名字,来获取其电话号码,达到拨号的目的。对一些非主要的功能,但又能提升用户体验的功能设计或者优化,这是我们做程序设计的第二步。

## 3.系统设计

本部分是程序设计的第二个环节,了解了客户需求后,接下来就是分析客户需求,分别通过对概要、数据结构、头文件部分的设计来实现达到用户满意的目的

### 3.1 概要设计

本部分设计主要从函数的功能这个角度考虑,根据用户需求分析,找到需要设计的模块,在这些模块的基础上一一进行设计。

做一个通讯录系统,至少包括以下四个板块:主函数板块,用户界面板块(即菜单板块),用户功能板块,数据输出板块。

1) 主函数板块的主要功能是为程序提供入口,通过 main 函数的不同选择进入到各功能板块。

2) 用户界面板块(即菜单板块)主要给用户提供一个界面,使用户能够进行操作,一般用自定义的 menu 函数实现。

3) 用户功能板块是通过需求分析后得出的用户所需要实现的功能,例如:增加功能 in,删除功能 del,修改功能 modify,查看功能 order,拨号功能 total。

4) 数据输出板块通过定义显示函数 show,由于各功能输出要求不同,因此设计了不同的输出函数,通过加数字编号来区分。

在进行大程序设计时,我们应当根据需求,把一个

大板块划分成许多个小版块,再根据每一版块的不同要求分别进行设计。

### 3.2 数据结构

根据项目的不同,我们研究的主体也不相同,数据结构往往是根据这个主体,来设计它的必要信息,将这一板块单独设计成一个数据类型,便于函数功能的操作。

以通讯录设计为例,本系统的研究主题是通讯录信息,它包括四个基本数据姓名、电话号码、分类(我们规定其类型有三类,分别为A办公类、B个人类、C商务类)、电子邮件,因此我们设计的数据结构应包含这四个数据。

示例如下:

```
struct Info{
char name[15];/*姓名*/
char tel[15];/*电话*/
char type[15];/*分类*/
char mail[20];/*电子邮件*/
};struct Info li[50];
```

### 3.3 头文件部分

头文件部分,使用#include进行包含,往往和项目的需求有关,通过分析需要用到C语言的哪些功能,才能找到本程序所需要的头文件,在刚开始设计时,我们往往会漏掉一些头文件,在进行后面的编写时,当发现需要使用头文件时,可以立刻回到头文件部分进行补充。

另外,当我们发现我们重复再使用一些比较长的格式、变量等时,可以返回头文件部分使用#define进行简化定义,将复杂的東西简单化。

```
void menu(){
system("cls");
printf("\n\n\t *****菜 单*****\n");//
菜单
printf("\t*          1.增加信息
*\n");
printf("\t*          2.删除信息
*\n");
printf("\t*          3.修改信息
*\n");
```

一个通讯录所需要的头文件,除了最基本的stdio.h头文件以外,我们还会用到其他的头文件,在编写到一定程度后,我们发现很多地方运用了同一格式或复杂变量,于是可以返回使用#define进行简化定义。

示例如下:

```
#include<stdio.h>
#include<stdlib.h>
#include<dos.h>
#include<string.h>
#include<conio.h>
#define LEN sizeof(struct Info)
#define FORMAT
"%-5d%-15s%-15s%-15s%-20s\n"
#define DATA
i+1,li[i].name,li[i].tel,li[i].type,li[i].mail
```

## 4.功能设计

本部分是程序设计的第三步,在打好了前面的基础后,这一步可以被划分成主程序和其他功能程序设计,可以同时由多人来设计。

### 4.1 主程序与菜单函数设计

主程序,即main函数,其主要功能是为程序提供入口,通过main函数的不同选择进入到各功能板块,在设计本版块时,往往配合菜单函数一并进行设计,主函数是用户无法看见的部分,而菜单函数是用户的界面,因此主函数设计应围绕用户界面进行设计。

以通讯录设计为例,根据用户需求,我们将预期需要的功能函数一一列举,先设计出一个菜单,即先设计用户界面。示例如下:

```
printf("\t*          4.查看信息
*\n");
printf("\t*          5.拨号
*\n");
printf("\t*          6.退出
*\n");
printf("\t
*****
*****\n");
printf("\t\t\t 请输入数字(1-6)并按回车键:");
}
```

其次, 根据我们设计好的菜单, 对菜单上的每一条功能逐一进行设计, 并将其入口放入主函数, 通过循环来设计菜单的刷新。

```
int main(){ /*主函数*/
    int n;
    menu();
    scanf("%d",&n);/*输入选择功能的编号*/
    while(n){
switch(n){
case 1: in();break;/*输入*/
case 2: del();break;/*删除*/
case 3: modify();break;/*修改*/
case 4: order();break;/*查看*/
case 5: total();break;/*拨号*/
void order(){
    FILE *fp;
    int snum,i,m=0;
    char ch[2];
    if((fp=fopen("data.txt","rb"))==NULL){
printf("can not open\n");return;}
    while(!feof(fp)) if(fread(&li[m],LEN,1,fp)==1)
m++;
```

根据每个功能的不同要求, 单独对同一文件进行操作, 各操作之间互不干扰, 每一个功能组合起来之后, 就会变成小程序, 一个个小程序组合起来之后便成了大程序。

### 4.3 输出函数设计

与其他语言形式相比, C 语言在输入以及输出的方式上具有一定的新颖性。[5]因此输出板块设计也尤其重要, 前面的板块, 除了菜单板块以外, 对于用户来说都是不可见的, 只有看到了最后的输出, 用户才能知道你的程序是否完成。

在通讯录设计中, 由于不同功能函数的输出要求不同, 编写出了四个大体相同但有区别的输出函数。这四个函数大体一致, 仅仅是根据不同的要求, 有不同的输出。

以下为其中一个输出函数示例:

```
void show2(char type1[])
{ FILE *fp;
    int i,m=0;
```

```
case 6: exit(0);break;/*退出*/
default:break;
    }
    getch();
    menu();/*执行完功能再次显示菜单界面*/
    scanf("%d",&n);
    }
}
```

### 4.2 功能函数设计

功能函数就是根据需求实现的通讯录功能, 来对每一个板块进行设计, 使得用户从主函数接口进入功能函数后, 能后实现其预期效果。

查看功能示例:

```
fclose(fp);
if(m==0) {printf("no record!\n");return;}
printf("请输入 A、B、C 选择要查看的类别:\n 办
公类\n 个人类\n 商务类\n");
char d[15];
scanf("%s",d);
show2(d);
}
fp=fopen("data.txt","rb");
while(!feof(fp))
{
if(fread(&li[m],LEN,1,fp)==1)
m++;
}
fclose(fp);
printf("姓名      电话号码      \t\n");
for(i=0;i<m;i++){
if(!strcmp(type1,li[i].type))printf("%-15s%-15s\n",li[
i].name,li[i].tel);
/*将信息按指定格式打印*/
}
}
```

### 5.结论

一个大程序的设计, 离不开一个一个小程序, 小程序的设计, 离不开一个一个板块的设计, 板块的设计, 离不开一个一个函数的设计, 因此, C 语言程序设计可

以简化成每一个函数的设计和组合问题,将一个复杂的问题简单化是 C 语言程序设计的思路,只有把大块划分成小块,才有办法逐一击破。一个程序的设计,必须从用户的需求的角度来考虑,必须把客户放在第一位,并且在设计时要尽可能地为用户做优化。在编写代码的过程中,多写注释也是一种很好的编程习惯,往往大程序设计是由多人合作完成的,因此多写注释可以方便团队之间的交流合作。

### 参考文献

- [1]曹亚萍.浅析学习 C 语言的必要性及学习方法[J].计算机产品与流通,2018(12):221
- [2]莫玲,姚屏.非计算机专业 C 语言教学方法探讨[J].教育教学论坛,2019(08):191-192.
- [3]王立涛.以用户为中心的应用系统分析与设计[D].大连海事大学,2015
- [4]沈鑫,陈华山.关于软件需求分析的理解[J].现代信息技术,2018,2(11):13-14+17.
- [5]向桂玲.分析计算机中 C 语言的应用特点[J].科技风,2018(15):81.

### 作者简介

第一作者:陈金鹏(1998-2-),男,汉,四川成都,本科,四川大学锦城学院,研究方向:软件工程;

第二作者(通讯作者):鲍正德(1989-7-),男,汉,黑龙江哈尔滨,研究生,四川大学锦城学院,研究方向:电子商务。