# 项目需求文档

**目录**

[一、项目背景与需求分析](#_Toc12577_WPSOffice_Level1) [1](#_Toc12577_WPSOffice_Level1)

[1.研究项目](#_Toc22993_WPSOffice_Level2) [1](#_Toc22993_WPSOffice_Level2)

[2. 项目分析背景](#_Toc27908_WPSOffice_Level2) [1](#_Toc27908_WPSOffice_Level2)

[3. 相关背景数据资料](#_Toc21141_WPSOffice_Level2) [2](#_Toc21141_WPSOffice_Level2)

[4. 项目需求](#_Toc1682_WPSOffice_Level2) [2](#_Toc1682_WPSOffice_Level2)

[二、本次需求的目的及功能列表](#_Toc22993_WPSOffice_Level1) [2](#_Toc22993_WPSOffice_Level1)

[1.需求分阶段](#_Toc13820_WPSOffice_Level2) [3](#_Toc13820_WPSOffice_Level2)

[2. 需求的方向](#_Toc6475_WPSOffice_Level2) [3](#_Toc6475_WPSOffice_Level2)

[3. 需求功能或页面](#_Toc24487_WPSOffice_Level2) [3](#_Toc24487_WPSOffice_Level2)

[三、项目模块关系](#_Toc27908_WPSOffice_Level1) [3](#_Toc27908_WPSOffice_Level1)

[1.分工表](#_Toc11492_WPSOffice_Level2) [3](#_Toc11492_WPSOffice_Level2)

[2.页面用户使用情况](#_Toc20652_WPSOffice_Level2) [5](#_Toc20652_WPSOffice_Level2)

[四、功能详细介绍](#_Toc21141_WPSOffice_Level1) [5](#_Toc21141_WPSOffice_Level1)

[五、简单的用列介绍](#_Toc1682_WPSOffice_Level1) [5](#_Toc1682_WPSOffice_Level1)

[1.关键的用例及代码介绍](#_Toc15752_WPSOffice_Level2) [5](#_Toc15752_WPSOffice_Level2)

[2. 重点关注点](#_Toc15945_WPSOffice_Level2) [10](#_Toc15945_WPSOffice_Level2)

### 一、项目背景与需求分析

#### 1.研究项目

制作一个小程序（软件），当用户输入想购买的商品及数量时，计算出花费最少的购物方案及费用。所谓购物方案，指从哪个网站的哪家店铺购买多少什么商品。考虑京东、淘宝、天猫、沃尔玛线上等人们常用的电子商务网站。

#### 项目分析背景

该研究项目类似于最少费用购物问题：某商店中每种商品都有一个价格。例如，一朵花的价格是2元；一个花瓶的价格是5 元。为了吸引更多的顾客，商店提供了特殊优惠价。特殊优惠商品是把一种或几种商品分成一组。并降价销售。例如:3朵花的价格不是6而是5元 ；2个花瓶加1朵花是10 元，设计一个算法，计算出某顾客所购商品应付的最少费用。

#### **相关背景数据资料**

要计算最少费用购物问题：首先要设计设计一个算法，计算出某一顾客所购商品应付的最少费用。对于给定欲购商品的价格和数量，以及优惠商品价，编程计算所购商品应付的最少费用。数据输入：由文件input.txt提供欲购商品数据。此问题看起来除了进行穷举求最小购物费用。

#### **项目需求**

当今社会是一个多元化的社会，电子产品和网络社会的发展，给我们的生活方式带来了便捷。特别是近年来网购平台的出现，加速了经济生活的快速发展，越来越多的群众追求，快速便捷的经济实惠的生活，而制作一个小程序（软件），当用户输入想购买的商品及数量时，花费最少的费用成为人们逐渐追求的社会问题。也为各人民群众带来巨大便利。

### 二、本次需求的目的及功能列表

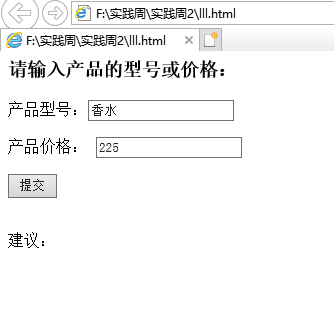
#### 1.需求分阶段

问题描述，设计算法，代码实现，数据输入，数据输出。

#### 需求的方向

输入产品的型号或价格，在我们所已有的数据中，输入其中的一类，提交，可以得到我们所要的最少费用的方案。

#### 需求功能或页面

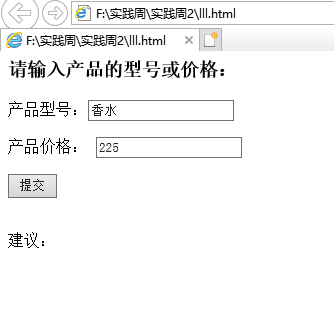


### 三、项目模块关系

#### 1.分工表

|  |  |  |
| --- | --- | --- |
| 分工表 | | |
| **名称** | **姓名** | **分工** |
| 组长 | 黄长连 | 查资料、写代码、计算代码复杂性、分工表 |
| 组员 | 张梅华 | 设计算法、写需求文档 |
| 组员 | 梁小娇 | 问题建模、总结遇到的问题和困难形成文档 |
| 组员 | 罗晓欣 | 测试、测试报告、写说明文档 |
| 组员 | 黄海琴 | 设计实验文档、写总结报告 |

#### 2.页面用户使用情况



### 四、功能详细介绍

当用户输入想购买的商品及数量时，花费最少的费用成为人们逐渐追求的社会问题。也为各人民群众带来巨大便利。

### 五、简单的用列介绍

#### 1.关键的用例及代码介绍

最少费用购物问题：

#include<stdio.h>  
#include<string.h>  
#define maxb 6 //所购商品种类数(0<=B<=5)  
#define maxs 100//优惠商品种类数(0<=S<=99)  
int purch[maxb][2];//存放预购商品数据  
int offer[maxs][maxb];//存放优惠商品价数据  
int product[maxb];  
int num[1000];  
int cost[maxb][maxb][maxb][maxb][maxb];  
int b,s;//b件商品，s种组合  
//初始化  
void init()  
{  
int i,j,k;  
int code,t,p;//code表编号，t表示优惠商品组合  
memset(offer,0,sizeof(offer));  
    memset(purch,0,sizeof(purch));  
    memset(product,0,sizeof(product));  
printf("请输入商品件数:");  
scanf("%d",&b);  
for(i=1;i<=b;i++)  
{  
scanf("%d %d %d",&code,&purch[i][0],&purch[i][1]);//code表示商品的编码；存放购买该种商品的总数；该商品的正常单价  
num[code]=i;//商品的编号num[2]=1,num[8]=2  
}  
   printf("请输入优惠组合数目:");  
   scanf("%d",&s);  
   for(i=1;i<=s;i++)  
   {  
   scanf("%d",&t);  
   for(j=1;j<=t;j++)  
   {  
   scanf("%d %d",&code,&p);  
   offer[i][num[code]]=p;//第i种优惠组合中编号为code的商品的数量   
   }  
    scanf("%d",&offer[i][0]);//利用0下标存第i种优惠组合的总价格   
   }  
}  
void mincost()  
{  
int i,min;  
int A,B,C,D,E;  
min=0;  
    for(i=1;i<=b;i++)  
{  
min=min+(product[i]\*purch[i][1]);//正常单价购买  
}  
//遍历S种优惠组合  
   for(i=1;i<=s;i++)  
   {    //五种产品减去采用优惠组合的数量  
   A=product[1]-offer[i][1];  
   B=product[2]-offer[i][2];  
       C=product[3]-offer[i][3];  
       D=product[4]-offer[i][4];  
       E=product[5]-offer[i][5];  
 if(A>=0&&B>=0&&C>=0&&D>=0&&E>=0&&min>(cost[A][B][C][D][E]+offer[i][0]))  
 {  
    min=cost[A][B][C][D][E]+offer[i][0];  
 }  
}  
   cost[product[1]][product[2]][product[3]][product[4]][product[5]]=min;  
}  
  
void dp(int i)  
{  
int j;  
if(i>b){  
mincost();  
return;  
}  
for(j=0;j<=purch[i][0];j++)  
{  
  
product[i]=j;  
  
dp(i+1);  
}  
}  
int main(int argc, char \*argv[])  
{  
int A,B,C,D,E;  
init();  
dp(1);  
A=purch[1][0];  
B=purch[2][0];  
C=purch[3][0];  
D=purch[4][0];  
E=purch[5][0];  
printf("%d\n",cost[A][B][C][D][E]);  
return 0;  
}

#### 重点关注点

  最终结果要为的最少的费用，给客户提供一种最少费用的购物方案。