**实验报告**

**课程名称：算法设计与分析**

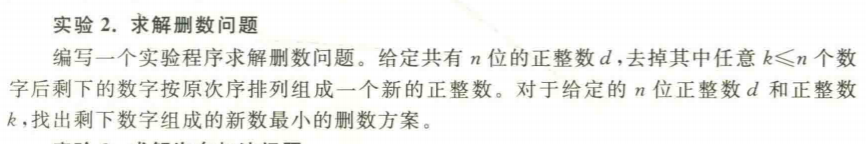
**专业班级：大数据182**

**学号姓名：3180439031陈佳婧**

**实验日期：2020年11月23日**

**第七章上机实验题2： 求解删数问题**

1. **实验目标和要求：**



**2. 实验环境：（操作系统、语言、编译工具…）WIN10、C++、Clion**

**3. 关键问题及解决思路：**

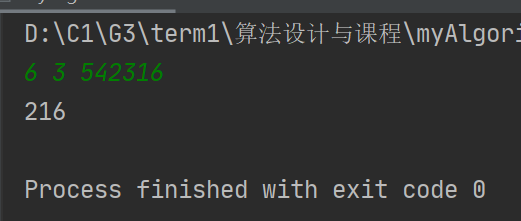
**贪心+递归+标记+下标记录。**

**4. 程序流程：**

**5. 完整程序代码：**

#include <iostream>  
#include <cstring>  
#include <algorithm>  
using namespace std;  
#define MAXN 100  
int ok[MAXN];  
int n=6,k=3,r=3;  
int cnt=0;  
void choose(char a[],int left,int right){  
 if(cnt==r)return;  
 if(left>right)return;  
 if(right<left)return;  
 char mins='9';  
 int i,tmp;  
 for(i=left;i<=right;i++){  
 if(a[i]<=mins){  
 mins=a[i];  
 tmp=i;  
 }  
 }  
 ok[tmp]=1;  
 cnt++;  
 choose(a,tmp+1,right);  
 choose(a,left,tmp-1);  
}  
int main() {  
 int d;//=542316;//216  
 //分析  
 /\* cnt=0  
 \* r=n-k  
 \* if cnt==r,ok  
 \* 找到最小的,cnt++  
 \* 在最小的左边，找最小，cnt++  
 \* 左边没了，找右边最小，cnt++  
 \*/  
 cin>>n>>k>>d;  
 string str=to\_string(d);  
 char demo[20] = "";  
 strcpy(demo,str.c\_str());  
 memset(ok,0,sizeof(ok));  
 choose(demo,0,n-1);  
 for(int i=0;i<n;i++){  
 if(ok[i])cout<<str[i];  
 }cout<<endl;  
 return 0;  
}

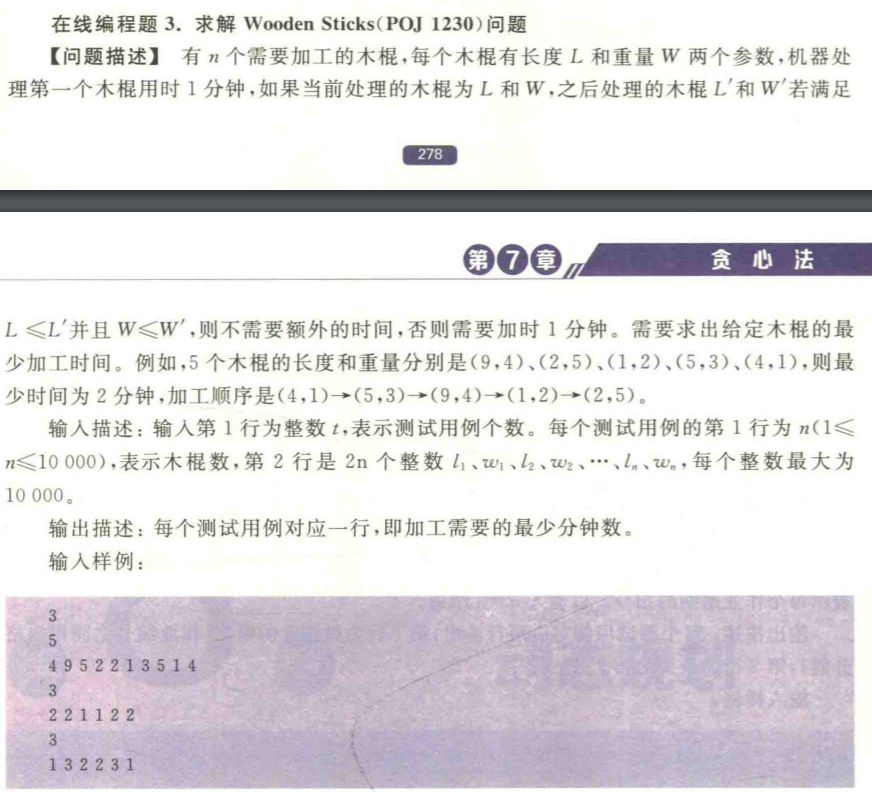
**6. 运行结果展示：**

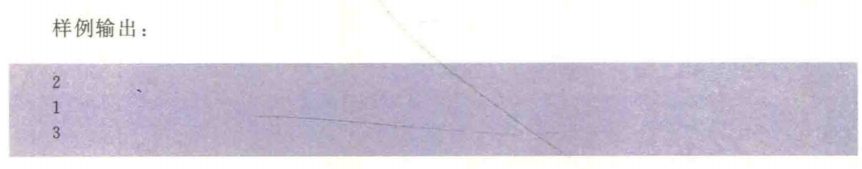


**7. 实验体会（可选）：**

**第七章在线编程题3： 求解Wooden Sticks(POJ 1230)问题**

**1.实验目标和要求：**





**2. 实验环境：（操作系统、语言、编译工具…）WIN10、C++、Clion**

**3. 关键问题及解决思路：**

Sort一遍，转化为序列包含几个最长上升子序列（贪心+二分）问题

**4. 程序流程：**

**5. 完整程序代码：**

#include <iostream>  
#include <cstring>  
#include <algorithm>  
using namespace std;  
const int MAXN=10005;  
  
struct Stick{  
 int l;  
 int w;  
}s[MAXN];  
bool cmp(Stick x1,Stick x2){  
 if(x1.l==x2.l){  
 return x1.w<x2.w;  
 }else return x1.l<x2.l;  
}  
  
int main() {  
 int n,t;  
 cin>>t;  
 int len[MAXN];  
 while(t--){  
 memset(len,0,sizeof(len));  
 cin>>n;  
 for(int i=1;i<=n;i++){  
 cin>>s[i].l>>s[i].w;  
 }  
 sort(s+1,s+n+1,cmp);  
 //转化为求包含几个最长上升子序列  
 int cnt=0,ans=0;  
 while(cnt<n){  
 ans++;  
 int maxl=0;  
 for(int i=1;i<=n;i++){  
 if(s[i].w==0)continue;  
 if(len[maxl]<=s[i].w){  
 len[++maxl]=s[i].w;  
 cnt++;  
 s[i].w=0;  
 }else{  
 int flag=lower\_bound(len,len+1+n,s[i].w)-len;  
 len[flag]=s[i].w;  
 }  
 }  
 }  
 cout<<ans<<endl;  
 }  
 return 0;  
}

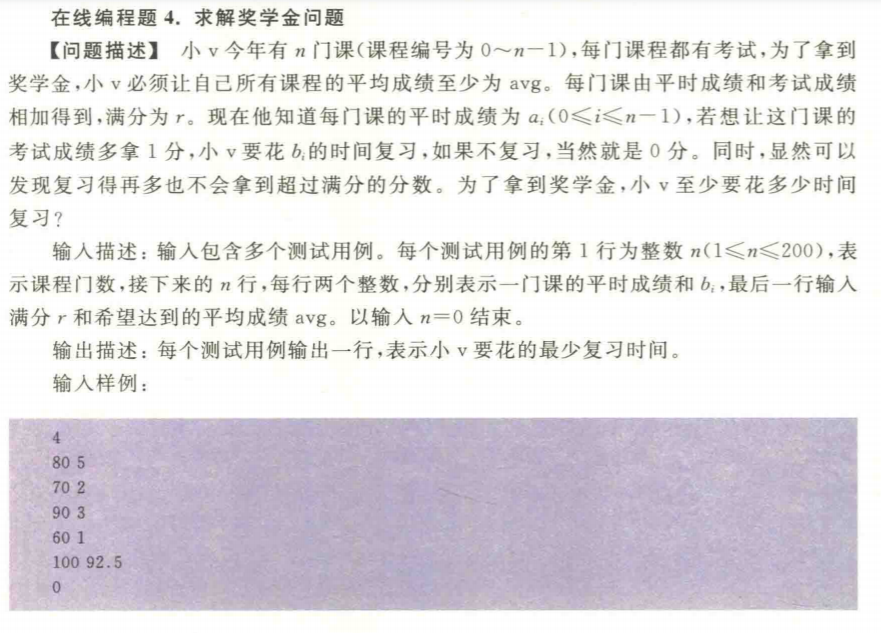
**6. 运行结果展示：**



**7. 实验体会（可选）：**

**第七章在线编程题4： 求解奖学金问题**

**1.实验目标和要求：**





改为100

**2. 实验环境：（操作系统、语言、编译工具…）WIN10、C++、Clion**

**3. 关键问题及解决思路：**

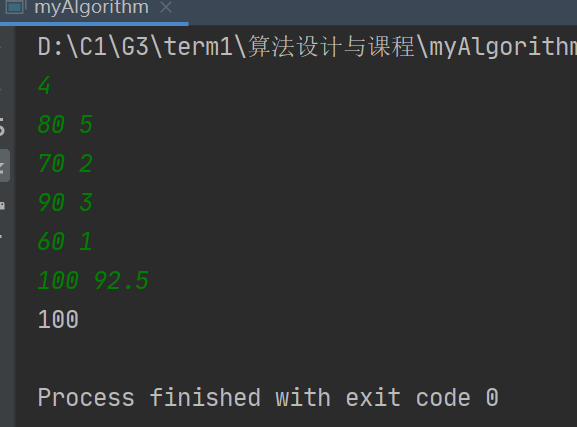
**贪心。**

**4. 程序流程：**

**5. 完整程序代码：**

#include <iostream>  
#include <cstring>  
#include <algorithm>  
using namespace std;  
#define MAXN 300  
int n,r;//n=4,r=100;  
double avg;//=92.5;  
  
bool cmp(int \*p,int \*q){  
 if(p[1]==q[1]){  
 return p[0]<q[0];//从小到大  
 }else return p[1]<q[1];//从小到大  
}  
//0,1  
//60,1  
int main() {  
 int sum=0;  
 int a[MAXN];//={80,70,90,60};  
 int b[MAXN];//={5,2,3,1};  
 cin>>n;  
 for(int i=0;i<n;i++){  
 cin>>a[i]>>b[i];  
 }  
 cin>>r>>avg;  
 int \*\*in=new int\*[n];  
 for(int i=0;i<n;i++){  
 in[i]=new int[2];  
 in[i][0]=a[i];  
 in[i][1]=b[i];  
 }  
 for(int i=0;i<n;i++)sum+=in[i][0];  
 int left=(int)(avg\*n)-sum;  
 sort(in,in+n,cmp);  
 int cost=0;  
 for(int i=0;i<n;i++){  
 if(r-in[i][0]<left){  
 cost+=(r-in[i][0])\*in[i][1];  
 left-=(r-in[i][0]);  
 }else{  
 cost+=left\*in[i][1];  
 break;  
 }  
 }  
 cout<<cost<<endl;  
 return 0;  
}

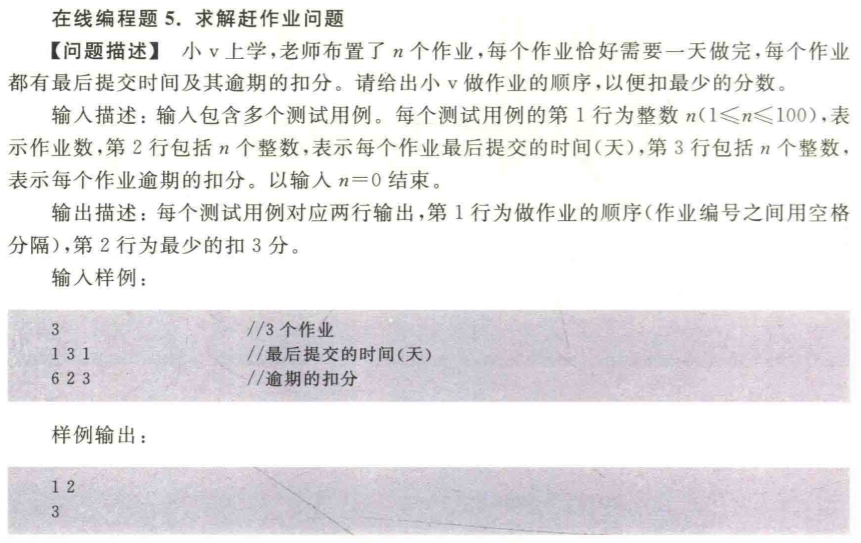
**6. 运行结果展示：**



**7. 实验体会（可选）：**

**第七章在线编程题5： 求解赶作业问题**

**1.实验目标和要求：**



**2. 实验环境：（操作系统、语言、编译工具…）WIN10、C++、Clion**

**3. 关键问题及解决思路：**

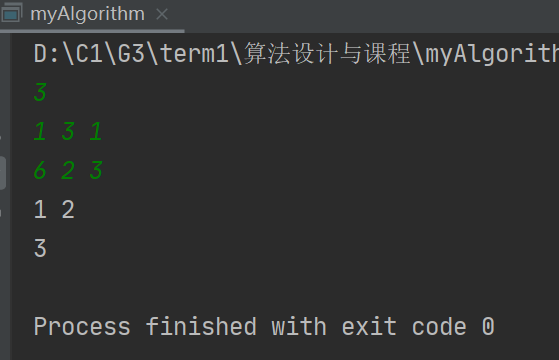
**贪心+标记。**

**4. 程序流程：**

**5. 完整程序代码：**

#include <iostream>  
#include <cstring>  
#include <algorithm>  
using namespace std;  
#define MAXN 105  
int n;  
struct Work{  
 int id;  
 int dl;  
 int cost;  
}works[MAXN];  
bool cmp(Work x1,Work x2){  
 if(x1.dl==x2.dl){  
 return x1.cost>x2.cost;//从大到小  
 }else return x1.dl<x2.dl;//从小到大  
}  
int visd[MAXN];  
int main() {  
 memset(visd,0,sizeof(visd));  
 cin>>n;  
 for(int i=0;i<n;i++){  
 cin>>works[i].dl;  
 works[i].id=i+1;  
 }  
 for(int i=0;i<n;i++){  
 cin>>works[i].cost;  
 }  
 sort(works,works+n,cmp);  
 int ans=0;  
 for(int i=0;i<n;i++){  
 if(!visd[works[i].dl]){  
 if(i!=n-1)cout<<works[i].id<<" ";  
 else cout<<works[i].id<<endl;  
 visd[works[i].dl]=1;  
 }else{  
 ans+=works[i].cost;  
 }  
 }  
 cout<<ans<<endl;  
 return 0;  
}

**6. 运行结果展示：**



**7. 实验体会（可选）：**