**《算法设计与分析》实验报告**

# 实验三 贪心算法

|  |  |  |  |
| --- | --- | --- | --- |
| **报告书** | | | |
| 姓名 | 吴宇敖 | 指导教师 |  |
| 学号 | 176001752 | 日 期 | 2020.4.25 |
| 班级 | 17计算机软件一班 |  |  |
| **实验内容** | | | |
| **[要求先在OJ上提交通过]**  请给出AC截图和代码；无需说明  1) 区间问题1[OJ]  2) 种树[OJ]  3）智力大冲浪[OJ] | | | |
| **实验目的** | | | |
| 1）按时完成堂练三题  2）熟练掌握贪心算法 | | | |
| **实验过程和步骤** | | | |
| **以下仅供参考，请在报告中删除**  **【实验题目】**  **题目描述：**  **给出n个区间的起点和终点，求最少使用其中多少个区间可以将所有区间所在的区域完全覆盖。（测试的数据确保这1点）**  **输入要求：**  **第1行一个整数n，表示n个区间；**  **第2行开始n行，每行2个整数，表示一个区间范围。**  **类似[1,4][5,6]被认为是覆盖了[1,6]。**  **输出要求：**  **从起点开始，按区间先后顺序，输出选中的区间。所选的区间应尽可能向终点扩展。**  **【程序代码及注释】**  #include<iostream>  #include<algorithm>  using namespace std;  struct node{  int left;  int right;  }a[1000];  int cmp(node a,node b){  return a.left<b.left;  }  int main()  {  int n,k,x,m;  int res[100]={0};  cin>>n;  for(int i=0;i<n;i++)  cin>>a[i].left>>a[i].right;  sort(a,a+n,cmp);//排序，按恪照左边的数，从小到大排序  k=a[0].right;  res[0]=1;  for(int i=1;i<n;i++)//选择一开始相同左节点大小的数中，右节点最大的  if(a[i].left==a[0].left)  {  k=a[i].right;  res[0]=i+1;  }  else  break;  for(int i=res[0];i<n;i++)//循环  {  for(int j=i,m=0;j<n;j++)//选择下一个左节点小于等于选出点的右节点加一并且右节点大于选出点的右节点  {  if(a[j].left<=k+1&&a[j].right>k)  if(m==0)  {  x=j;  m++;  }  else if(a[j].right>a[x].right)  x=j;  }  k=a[x].right;  res[x]=x+1;//记录需要使用的区间  }  for(int i=0;i<n;i++)//输出  if(res[i]!=0)  cout<<a[res[i]-1].left<<" "<<a[res[i]-1].right<<endl;  //system("pause");  return 0;  }  **【实验结果及遇到的问题】**    堂练后AC的截图：  QQ截图20200423103503  遇到的问题：  （1）一开始我使用了递归的方法，一直出现错误，之后我就把它推到，从新写过  （2）对于测试数据中6 9和7 9的数据选择出现了问题，答案是6 9，但是按照我写的代码是7 9，对于这两组数据：1 7、6 9、9 10和1 7、7 9、9 10都是可以完全覆盖，但是提交后是错误的，之后就在比较时多加了一个比较约束。  **【算法分析】**  先输入一组数据，对于第一个数先进行从小到大的排序，在选择最小的左节点对应的最远的右节点的那组数。之后选择下一个左节点小于等于选出点的右节点加一并且右节点大于选出点的右节点的数，选其中右节点最大的数，重复循环。  **【实验题目】**  **题目描述：**  **一条街的一边有几座房子。因为环保原因居民想要在路边种些树，路边的地区被分割成块，并被编号成1..N；每个部分为一个单位尺寸大小并最多可种一棵树，每个居民想在门前种些树并指定了三个号码B,E,T，这三个数表示该居民想在B和E之间最少种T棵树。当然，B≤E,居民必须记住在指定区不能种多于区域地块数的树，所以T≤E-B+l。居民们想种树的各自区域可以交叉。你的任务是求出能满足所有要求的最少的树的数量。**  **输入要求：**  **第一行包含数据N，区域的个数；**  **第二行包含H，房子的数目；**  **下面的H行描述居民们的需要:B E T。**  **输出要求：**  **输出能满足所有要求的最少的树的数。**  **【程序代码及注释】**  #include<iostream>  #include<algorithm>  #include<cstring>  using namespace std;  int tr[1000000]={0};  struct node{  int l,r,tree;  }a[1000000];  bool cmp(node c,node b){  return c.r<b.r;  }  int min(int a,int b){  if(a<=b)return a;  else return b;  }  int main(){  int m,n,res=0;  cin>>m>>n;  for(int i=0;i<n;i++){  cin>>a[i].l>>a[i].r>>a[i].tree;  }  sort(a,a+n,cmp);//排序  int count=0;  for(int i=0;i<n;i++){  for(int j=a[i].l;j<=a[i].r;j++){  if(a[i].tree==0)//不需要种树就退出  break;  if(tr[j]==1)//已经有树了，就将需要种的树减一  a[i].tree--;  }  for(int j=a[i].r;j>=a[i].l;j--){  if(a[i].tree==0)//不需要种树就退出  break;  if(tr[j]==0){//在j的位置种一棵树  a[i].tree--;  res++;  tr[j]=1;  }  }  }  cout<<res<<endl;;  //system("pause");  return 0;  }  **【实验结果及遇到的问题】**    已在堂练时AC  **【实验题目】**  **题目描述：**  **小伟报名参加中央电视台的智力大冲浪节目，本次挑战赛吸引了众多参赛者，主持人为了表彰大家的勇气，先奖励每个参赛者m元。先不要太高兴！因为这些钱还不一定都是你的！接下来主持人宣布了比赛规则：**  **首先，比赛时间分为n个时段(n≤500),它又给出了很多小游戏，每个小游戏都必须在规定期限ti前完成(1≤ti≤n)。如果一个游戏没能在规定期限前完成，则要从奖励费m元中扣去一部分钱wi，wi为自然数，不同的游戏扣去的钱是不一样的。当然，每个游戏本身都很简单，保证每个参赛者都能在一个时段内完成，而且都必须从整时段开始。主持人只是想考考每个参赛者如何安排组织自己做游戏的顺序。作为参赛者，小伟很想赢得冠军，当然更想赢取最多的钱！注意:比赛绝对不会让参赛者赔钱！**  **输入要求：**  **输入文共4行。**  **第1行为m，表示一开始奖励给每位参赛者的钱；**  **第2行为n，表示有n个小游戏；**  **第3行有n个数，分别表示游戏1到n的规定完成期限；**  **第4行有n个数，分别表示游戏1到n不能在规定期限前完成的扣款数。**  **输出要求：**  **输出仅1行，表示小伟能赢取最多的钱**  **【程序代码及注释】**  #include<iostream>  #include<algorithm>  #include<cstring>  using namespace std;  int num[500]={0},y;  struct node{  int t;  int m;  }a[1000];  int cmp(node a,node b){  return a.m>b.m;  }  int main(){  int n,m,res=0;  cin>>m>>n;  for(int i=0;i<n;i++){//输入  cin>>a[i].t;  }  for(int i=0;i<n;i++){  cin>>a[i].m;  }  sort(a,a+n,cmp);//按照价值从大到小排序  for(int i=0;i<n;i++){//让价值较高的活动，先占用时间  if(num[a[i].t]==0)  num[a[i].t]=1;  else  {  for(int j=a[i].t;j>0;j--)  if(num[j]==0)  {  num[j]=1;  break;  }  else  if(j==1)  res+=a[i].m;  }  }  cout<<m-res<<endl;  //system("pause");  return 0;  }  **【实验结果及遇到的问题】**    已在堂练时AC | | | |