**《算法设计与分析》实验报告**

# 实验一 递归与分治策略

|  |  |  |  |
| --- | --- | --- | --- |
| **报告书** | | | |
| 姓名 | 吴宇敖 | 指导教师 |  |
| 学号 | 176001752 | 日 期 | 2020.3.7 |
| 班级 | 17计算机软件一班 |  |  |
| **实验内容** | | | |
| **[要求先在OJ上提交通过]**  1）输入n个数，求其中第k小的数。  2)n个同学顺序围成一个圈，传递一个“不挂科”令牌，如果最后一次能传回第一个同学，则得到过的人不会挂科；每个同学只能传给左边或者右边的同学；现在从1号同学开始，第1次传递到1；第2次可以从1传递到2或n。问经过m次传递后，传回到1号同学的可能方式是多少种？ | | | |
| **实验目的** | | | |
| 1）了解到排序的方法，怎样可以较快的查找到第k个最小的数  2）了解到传递的规律，使用递归的方法来进行计算 | | | |
| **实验过程和步骤** | | | |
| **【实验题目】**  输入n个数，求其中第k小的数。  **【程序代码及注释】**  #include<iostream>  using namespace std;  int main()  {  int num[1000];  int n,k,l,m;  cin>>n>>k;//输入  for(int i=0;i<n;i++)//初始化  cin>>num[i];  for(int i=0;i<k;i++)//进行排序，将最小的数放置到第一位，一位一位下去直到到第k位?  {  l=i;  for(int j=i;j<n;j++)  {  if(num[l]>num[j])//标记最小的数是数组的第几位  l=j;  }  if(l!=i)  {  m=num[l];//交换位置  num[l]=num[i];  num[i]=m;  }  }  cout<<num[k-1]<<endl;  //system("pause");  }  **【实验结果及遇到的问题】（附输入输出）**    **【算法分析】**  本实验简单的使用了冒泡排序，将最小的数放置在第一位，将第二小的数放置在第二位，依次下去直到到达第k位。  **【实验题目】**  n个同学顺序围成一个圈，传递一个“不挂科”令牌，如果最后一次能传回第一个同学，则得到过的人不会挂科；每个同学只能传给左边或者右边的同学；现在从1号同学开始，第1次传递到1；第2次可以从1传递到2或n。问经过m次传递后，传回到1号同学的可能方式是多少种？  **【程序代码及注释】**  #include<iostream>  using namespace std;  int main()  {  int m,n,k=0;  cin>>m>>n;  int num[101][101]={0};  num[1][1]=1;//第一次传递是到第一个人，只有一种情况  num[2][2]=1;//第二次传递是到第2和n第个人  num[2][n]=1;  for(int i=3;i<=m;i++)//通过循环，计算第i次传递，到第几个人，分别有几种情况  for(int j=1;j<=n;j++)//每次传递都只可能通过左右两位传递，所以只要将左右两位的可能相加就行  {  if(j==n)  num[i][j]=num[i-1][j-1]+num[i-1][1];  else if(j==1)  num[i][j]=num[i-1][n]+num[i-1][j+1];  else  num[i][j]=num[i-1][j-1]+num[i-1][j+1];  }  cout<<num[m][1]<<endl;  //system("pause");  return 0;  }  **【实验结果及遇到的问题】（附输入输出）**    问题：  （1）一开始我使用了递归的方法来进行分析，写好代码之后发现很难将代码转化为递推的形式，而如果不进行转化，会导致数据量一旦变大时无法进行结果的计算。  （2）在使用递推时，我忘记了一个人传递一次这种情况  **【算法分析】**  本实验使用了递推的方法，由于传递从第一个人开始，并只能向两边传递，所以定义一个二维数组，保存传递次数和传递可能性。将第一次传递中第一个人记为1，之后每增加一次传递，每个人传递的可能性就是前一次传递时左右两个人传递可能性的相加。 | | | |