**实验报告**

**课程名称：算法设计与分析**

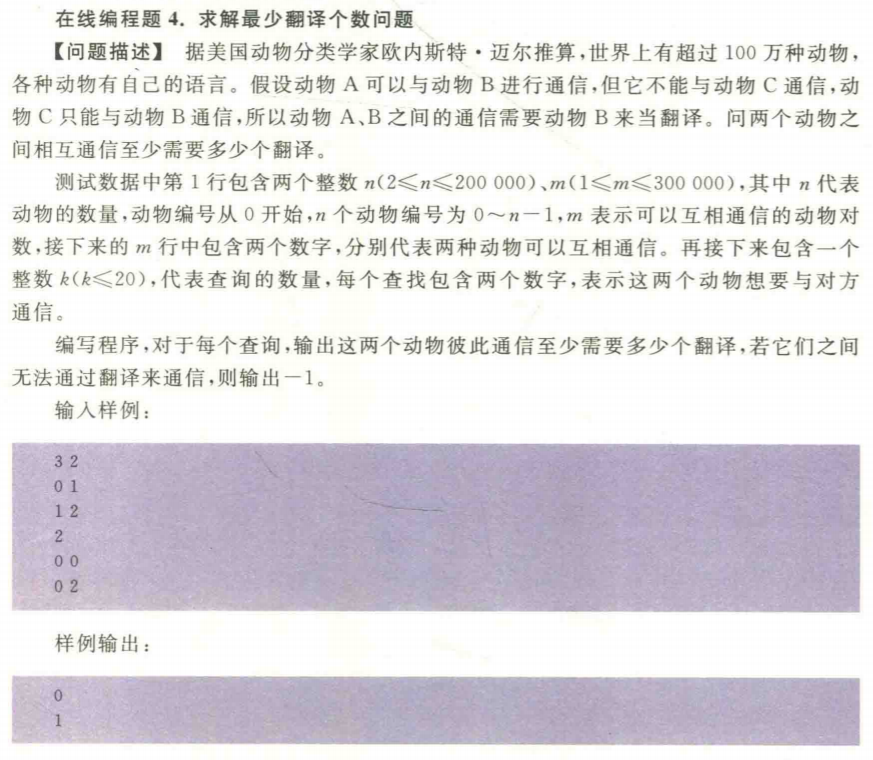
**专业班级：大数据182**

**学号姓名：3180439031陈佳婧**

**实验日期：2020年11月16日**

**第六章在线编程题4： 求解最少翻译个数问题**

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：**

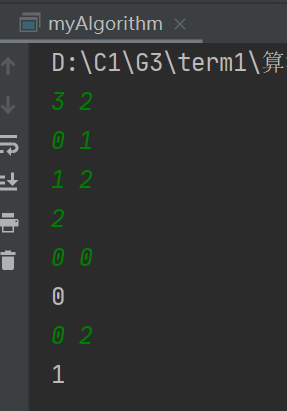
**优先队列式分支界限法，bfs。**

**4. 程序流程：**

**5. 完整程序代码：**

#include <iostream>  
#include <queue>  
#include <vector>  
#include <cstring>  
using namespace std;  
#define INF 0x3f3f3f3f  
#define MAXN 51  
int a[MAXN][MAXN];  
int n,v,m,k;  
int pre[MAXN];  
int dist[MAXN];  
struct NodeType{  
 int vno;  
 int length;  
 bool operator<(const NodeType&node)const{  
 return length>node.length;  
 }  
};  
void bfs(int v){  
 NodeType e{},e1{};  
 priority\_queue<NodeType> pqu;  
 e.vno=v;  
 e.length=0;  
 pqu.push(e);  
 dist[v]=0;  
 while(!pqu.empty()){  
 e=pqu.top();  
 pqu.pop();  
 for(int j=0;j<n;j++){  
 if(a[e.vno][j]<INF&&e.length+a[e.vno][j]<dist[j]){  
 dist[j]=e.length+a[e.vno][j];  
 pre[j]=e.vno;  
 e1.vno=j;  
 e1.length=dist[j];  
 pqu.push(e1);  
 }  
 }  
 }  
}  
void addEdge(int i,int j,int w){  
 a[i][j]=w;  
}  
int main() {  
 memset(dist,INF,sizeof(dist));  
 memset(a,INF,sizeof(a));  
 cin>>n>>m;  
 for(int i=0;i<m;i++){  
 int x,y;  
 cin>>x>>y;  
 addEdge(x,y,1);  
 }  
 cin>>k;  
 while(k--){  
  
 for(int i=0;i<n;i++){  
 a[i][i]=0;  
 }  
 v=0;  
 int s,e;  
 cin>>s>>e;  
 bfs(s);  
 if(dist[e]==0)cout<<0<<endl;  
 else if(dist[e]>n)cout<<-1<<endl;  
 else cout<<dist[e]-1<<endl;  
 }  
 return 0;  
}

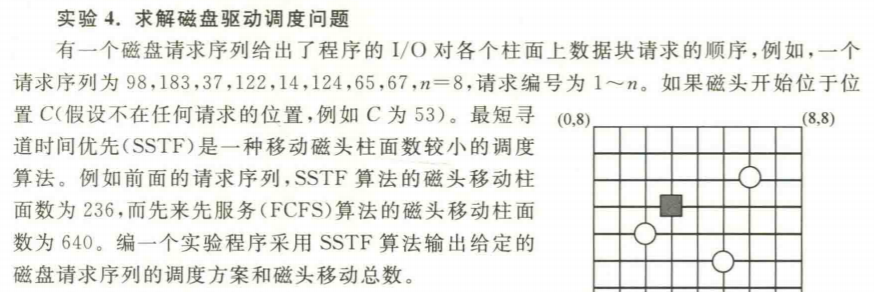
**6. 运行结果展示：**



**7. 实验体会（可选）：**

**第七章上机实验题4： 求解磁盘驱动调度问题**

**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：**

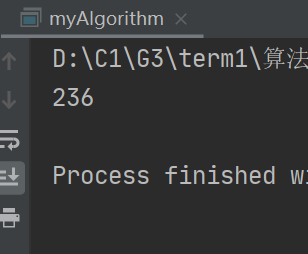
SSTF:最短寻道时间算法，算法本质是贪心，已知磁头的初始位置，则最先被处理就是距离磁头位置最近的进程，处理完成后再处理距离当前磁道最近的进程，直到所有的进程被处理。

**4. 程序流程：**

**5. 完整程序代码：**

#include <iostream>  
#include <cstring>  
#include <algorithm>  
using namespace std;  
#define MIN(x,y){x<y?x:y}  
#define INF 0x3f3f3f3f  
int main() {  
 int a[11]={-INF,53,98,183,37,122,14,124,65,67,INF};  
 int visd[11];  
 memset(visd,0,sizeof(visd));  
 int c=53,n=8,sum=0,cnt=0;  
 sort(a,a+n+2);  
 for(int i=1;i<n+1;i++){  
 if(c==a[i]&&!visd[i]){  
 visd[i]=1;  
 int l=i,r=i;  
 while(l--&&l>0){  
 if(!visd[l])break;  
 }  
 while(r++&&r<n+2){  
 if(!visd[r])break;  
 }  
 if(abs(c-a[l])<abs(c-a[r])){  
 sum+=abs(c-a[l]);  
 c=a[l];  
 i=l-1;  
 }else{  
 sum+=abs(c-a[r]);  
 c=a[r];  
 i=r-1;  
 }  
 }else{continue;}  
 }  
 cout<<sum<<endl;  
 return 0;  
}

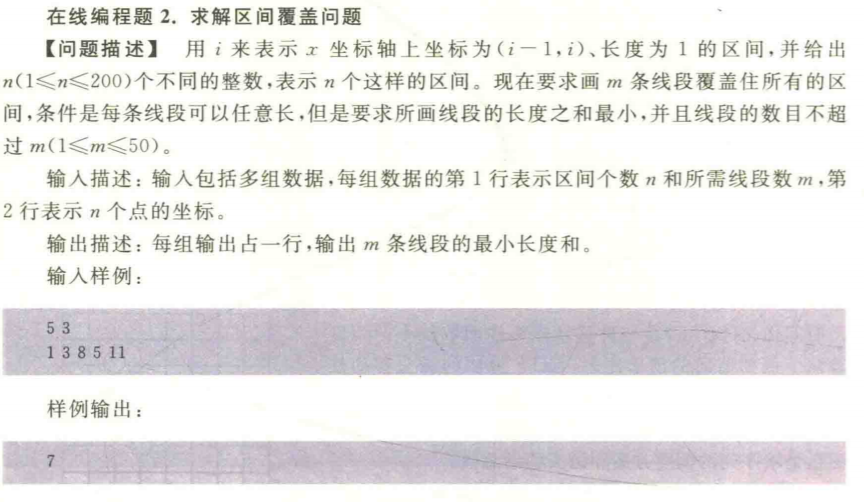
**6. 运行结果展示：**



**7. 实验体会（可选）：**

**第七章在线编程题2： 求解区间覆盖问题**

**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：**

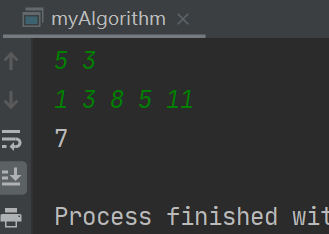
**贪心。**

**4. 程序流程：**

**5. 完整程序代码：**

#include <iostream>  
#include <cstring>  
#include <algorithm>  
using namespace std;  
#define MAXN 205  
int a[MAXN];//={1,3,8,5,11};  
int b[MAXN];//={0,0,0,0};  
int main() {  
 int n,m;//5,3  
 cin>>n>>m;  
 for(int i=0;i<n;i++){  
 cin>>a[i];  
 }  
 sort(a,a+n);  
 for(int i=0;i<n-1;i++){  
 b[i]=a[i+1]-a[i];  
 }  
 sort(b,b+n-1);  
 int sum=a[n-1]-a[0]+1,j=n-2;  
 for(int i=0;i<m-1;i++){  
 sum-=b[j--]-1;  
 }  
 cout<<sum<<endl;  
 return 0;  
}

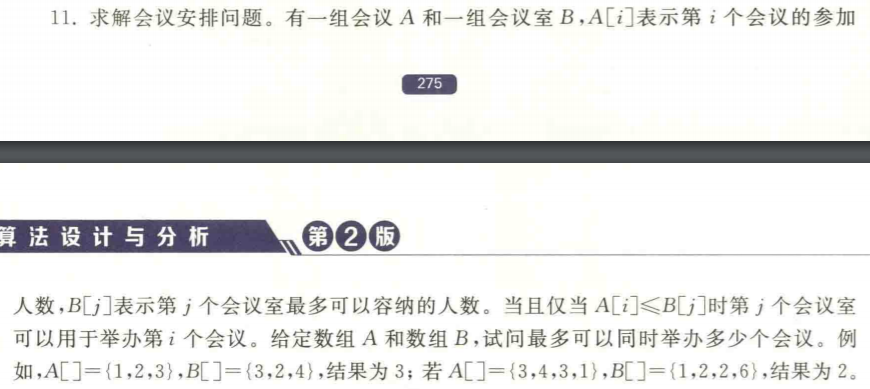
**6. 运行结果展示：**



**7. 实验体会（可选）：**

**第七章练习题11： 求解会议安排问题**

**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：**

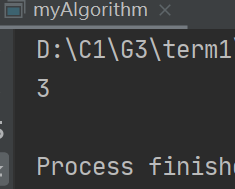
**贪心。**

**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 main() {  
 int a[MAXN]={1,2,3},b[MAXN]={3,2,4},n=3;  
 sort(a,a+n);  
 sort(b,b+n);  
 int cnt=0;  
 int ia=n,ib=n;  
 do{  
 if(a[ia-1]<=b[ib-1]){  
 cnt++;  
 ia--;  
 ib--;  
 }else{  
 ia--;  
 }  
 }while(ia&&ib);  
 cout<<cnt<<endl;  
 return 0;  
}

**6. 运行结果展示：**



**7. 实验体会（可选）：**