数据结构实践课报告

题目：稀疏矩阵运算器

班级：17052317 姓名：於文卓 学号17061833 完成日期：2018年11月19日

1. **需求分析**
2. 问题描述

很多涉及图上操作的算法都是以图的遍历操作为基础的。试着写一个程序，演示在联通的无向图上访问全部节点的操作

1. 基本要求

以邻接多重表为存储结构，实现BFS和DFS，用户指定起点，输出节点访问序列。

1. 测试数据：

　见课本187

1. **概要设计**
2. 数据结构

|  |
| --- |
| typedef int Status;  typedef enum {unvisited, isvisited} VisitIf;  typedef struct EBox  {  VisitIf mark;  int ivex;  int jvex;  struct EBox \*ilink;  struct EBox \*jlink;  } EBox;  typedef struct VexBox  {  char data[20];  EBox \*firstarc;  } VexBox;  typedef struct  {  VexBox adjmulist[MAX\_VERTEX\_NUM];  int vexnum;  int arcnum;  } AMLGraph; |

1. 基本操作

|  |
| --- |
| Status LocateVex(AMLGraph \*G, char \*v)  Status CreateGraph(AMLGraph \*G)  void DFS(AMLGraph \*G, int position)  Status DFSTraverse(AMLGraph \*G, int j)  void BFS(AMLGraph \*G, int position) |

1. **本程序包含3个模块：**

(1)主程序模块：

int main(){

输入起始位置；

选择操作；

处理命令；

输出结果；

return 0;

}

(2)数据结构——实现邻接多重表；

（3）运算单元模块——实现BFS和DFS；

各模块之间的调用关系如下：

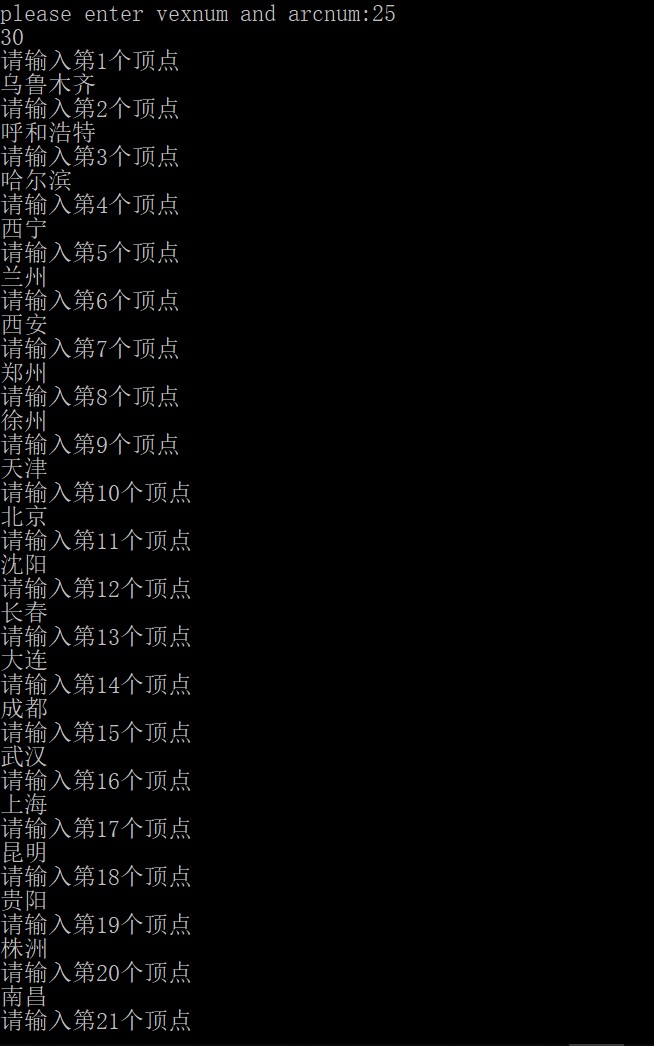
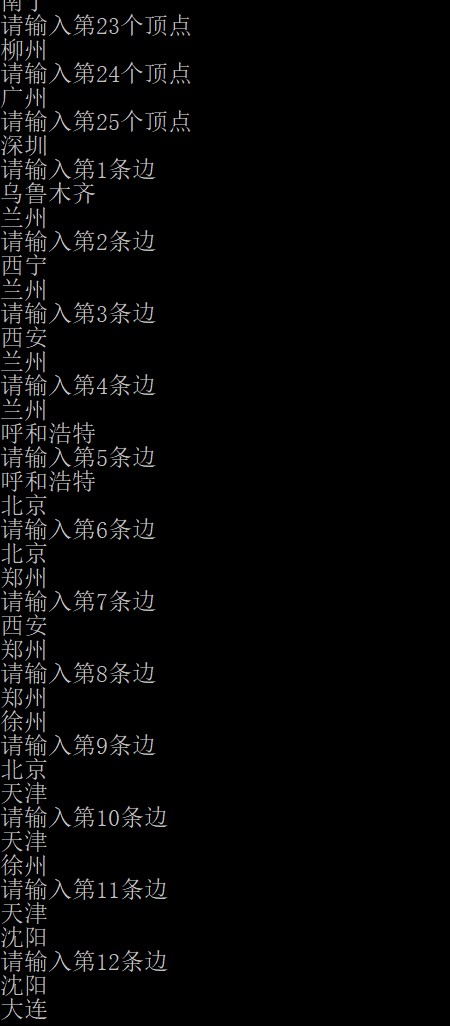
主程序模块 邻接多重表 遍历

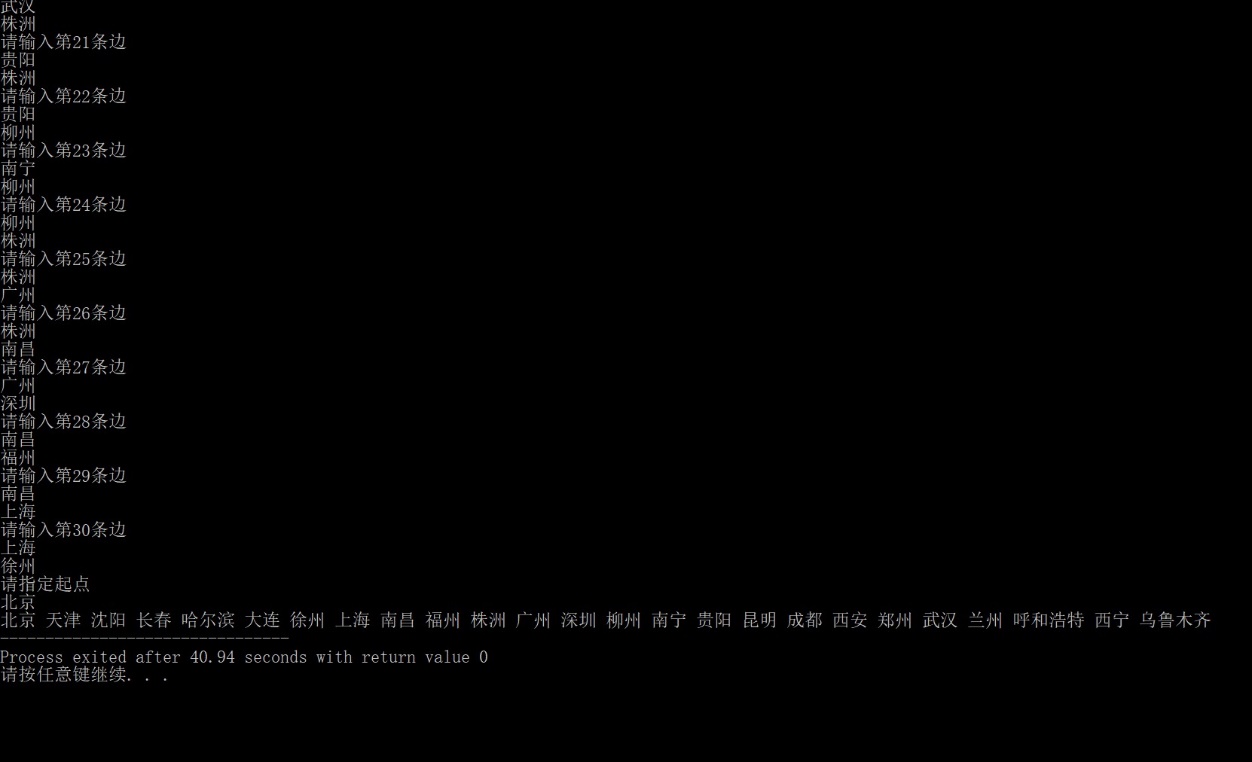
1. **运算单元模块部分实现代码**

|  |
| --- |
| **Status CreateGraph(AMLGraph \*G)**  **{**  **char va[200];**  **char vb[200];**  **EBox \*p;**  **int pos\_i;**  **int pos\_j;**  **printf("please enter vexnum and arcnum:");**  **scanf("%d%d", &(G->vexnum), &(G->arcnum));**  **for (int i = 0; i < G->vexnum; ++i)**  **{**  **printf("请输入第%d个顶点\n",i+1);**  **scanf("%s", G->adjmulist[i].data); //先给顶点表赋值**  **G->adjmulist[i].firstarc = NULL;**  **}**  **for (int k = 0; k < G->arcnum; ++k)**  **{**  **printf("请输入第%d条边\n",k+1 );**  **scanf("%s %s", va, vb); //输入两个有邻边的点**  **pos\_i = LocateVex(G, va); //找到顶点的下标**  **pos\_j = LocateVex(G, vb);**  **p = (EBox \*)malloc(sizeof(EBox));**  **p->ivex = pos\_i;**  **p->jvex = pos\_j;**  **p->mark = unvisited;**  **// 每次插入新节点都是在链表的表头进行**  **//这里没看懂要仔细看看**  **p->ilink = G->adjmulist[pos\_i].firstarc;**  **p->jlink = G->adjmulist[pos\_j].firstarc;**  **G->adjmulist[pos\_i].firstarc = p;**  **G->adjmulist[pos\_j].firstarc = p;**  **}**  **return OK;**  **}** |

|  |
| --- |
| **void DFS(AMLGraph \*G, int position)**  **{**  **visited[position] = TRUE;**  **EBox \*p;**  **int j;**  **printf("%s ", G->adjmulist[position].data);**  **p = G->adjmulist[position].firstarc;**  **while(p != NULL)**  **{**  **j = p->ivex == position ? p->jvex : p->ivex;**  **if(!visited[j])**  **DFS(G, j);**  **p = p->ivex == position ? p->ilink : p->jlink ;**  **}**  **}** |

1. **用户手册**
2. **本程序运行环境为win10操作系统，执行文件为main.exe**
3. **用户界面和测试数据如下**

****

****