电子科技大学信息与软件工程学院

**实 验 报 告**

学 号 2017221302006

姓 名 周玉川

（实验） 课程名称 数据结构

理论教师 刘勇国

实验教师 肖老师

**电子科技大学教务处制表**

**电 子 科 技 大 学**

**实 验 报 告（2）**

**学生姓名：** 周玉川  **学 号：2017221302006**  **指导教师： 刘勇国，肖老师**

**实验地点：**清水河基础实验大楼 **实验时间：2018.6.22**

**一、实验室名称：**学校实验中心软件实验室

**二、实验项目名称：**编程实现最小生成树Kruskal 算法

**三、实验学时：**4

**四、实验原理：**

Kruskal 算法是一种按照图中边的权值递增的顺序构造最小生成树的方法。其基本思想是：设无向连通网为G＝（V，E），令G 的最小生成树为T，其初态为T＝（V，{}），即开始时，最小生成树T 由图G 中的n 个顶点构成，顶点之间没有一条边，这样T 中各顶点各自构成一个连通分量。然后，按照边的权值由小到大的顺序，考察G 的边集E 中的各条边。若被考察的边的两个顶点属于T 的两个不同的连通分量，则将此边作为最小生成树的边加入到T 中，同时把两个连通分量连接为一个连通分量；若被考察边的两个顶点属于同一个连通分量，则舍去此边，以免造成回路，如此下去，当T 中的连通分量个数为1 时，此连通分量便为G 的一棵最小生成树。

如教材153页的图4.21(a)所示，按照Kruskal 方法构造最小生成树的过程如图4.21 所示。在构造过程中，按照网中边的权值由小到大的顺序，不断选取当前未被选取的边集中权值最小的边。依据生成树的概念，n 个结点的生成树，有n－1 条边，故反复上述过程，直到选取了n－1 条边为止，就构成了一棵最小生成树。

**五、实验目的：**

本实验通过实现最小生成树的算法，使学生理解图的数据结构存储表示，并能理解最小生成树Kruskal 算法。通过练习，加强对算法的理解，提高编程能力。

**六、实验内容：**

（1）假定每对顶点表示图的一条边，每条边对应一个权值；

（2）输入每条边的顶点和权值；

（3）输入每条边后，计算出最小生成树；

（4）打印最小生成树边的顶点及权值。

**七、实验器材（设备、元器件）：**

PC机一台，装有C语言集成开发环境。

**八、数据结构与程序：**

|  |
| --- |
| **//Kruskal算法 by Zhou Yuchuan**  **#include <bits/stdc++.h>**  **using namespace std;**  **#define MAXV 100**  **typedef struct**  **{**  **int u;**  **int v;**  **int weight;**  **}Edge;**  **int FindSet(int x,int Set[])**  **{**  **if (x==Set[x])**  **return x;**  **else Set[x]=FindSet(Set[x],Set);**  **}**  **bool mycompare(Edge e1,Edge e2)**  **{**  **return e1.weight<e2.weight;**  **}**  **void Kruskal(void)**  **{**  **int i,j,k,u1,v1,sn1,sn2;**  **char str[MAXV];**  **int Set[MAXV];**  **Edge E[MAXV\*10];**  **k=0;**  **int n,m;**  **printf("请输入顶点个数: \n");**  **scanf ("%d",&n);**  **m=n\*(n-1)/2;**  **printf("请输入%d个顶点标识(eg: abcde): \n",n);**  **while (getchar()!='\n');**  **gets(str);**  **//scanf("%s",str);**  **int w;**  **for (i=0;i<n;i++)**  **for (j=i+1;j<n;j++)**  **{**  **printf("边(%c,%c)权重为: ",str[i],str[j]);**  **scanf("%d",&w);**  **E[k].u=i;E[k].v=j;E[k].weight=w;**  **k++;**  **}**  **sort(E,E+m,mycompare);**  **for (i=0;i<n;i++)**  **Set[i]=i;**  **j=0;**  **printf("利用Kruskal算法得出最小生成树: \n");**  **for (k=0;k<n-1;j++)**  **{**  **u1=E[j].u;**  **v1=E[j].v;**  **sn1=FindSet(u1,Set);**  **sn2=FindSet(v1,Set);**  **if (sn1!=sn2)**  **{**  **printf("%c--->%c,%d\n",str[u1],str[v1],E[j].weight);**  **k++;**  **Set[sn1]=sn2;**  **}**  **}**  **}**  **int main(void)**  **{**  **int flag=1;**  **while (flag)**  **{**  **Kruskal();**  **printf("Program ended with exit code: ");**  **scanf ("%d",&flag);**  **}**  **return 0;**  **}** |

**九、程序运行结果：**

|  |
| --- |
| **（1）运行结果1：** |
| **结果评价:界面清晰，运行正常，符合要求。** |
| **（2）实验结果2:** |
| **结果评价：运行正常，能顺利输入。** |
| **（3）实验结果3：** |
| **结果评价：正确输出，顺利退出程序。** |

**十、实验结论：**

**试验成功，结论正确。**

**十一、总结及心得体会：**

1. 认真研究书本，虽然对新手来说有些困难，但是认真吸收后会受益无穷。
2. 平时多写代码，多练练，不能只在上机的时候才敲敲键盘，切勿眼高手低，没有简单的知识，只有勤快的学生。
3. 怀抱着兴趣去写代码。
4. 从第一次实验到现在，进步很大，代码敲的速度也快了，低级错误也极少犯，很欣慰，不辜负自己的努力。
5. 继续加油。