

数据结构

2022

# 实验报告

实验项目名称：图

班级：2021级6班

学号：2021302181138

姓名：伍旺旺

指导教师：沈志东

实验时间：2022 年 5 月 12 日

## 实验 8: 图

### 一、实验要求

(一) 实验题 5: 采用普里姆算法求最小生成树

(二) 实验题 10: 求有向图的简单路径

### 二、实验环境

硬件: 微型计算机

软件: Windows 操作系统、Microsoft Visual Studio Code

### 三、实验步骤及思路

(一) 实验题 5: 采用普里姆算法求最小生成树

- 题目分析

该程序包含如下函数:

$\text{Prim}(g, v)$ : 采用普里姆算法输出图  $g$  中从顶点  $v$  出发的一棵最小生成树

- 实验具体步骤

1. Prim 函数编写:

```

void Prim(MatGraph g,int v)
{
    int lowcost[MAXV],min,n=g.n;
    int closest[MAXV],i,j,k;
    for (i=0;i<n;i++)
    {
        lowcost[i]=g.edges[v][i];
        closest[i]=v;
    }
    for (i=1;i<n;i++)
    {
        min=INF;
        for (j=0;j<n;j++)
            if (lowcost[j]!=0 && lowcost[j]<min)
            {
                min=lowcost[j]; k=j;
            }
        printf(" 边(%d,%d)权为:%d\n",closest[k],k,min);
        lowcost[k]=0;
        for (j=0;j<n;j++)
            if (g.edges[k][j]!=0 && g.edges[k][j]<lowcost[j])
            {
                lowcost[j]=g.edges[k][j];
                closest[j]=k;
            }
    }
}

```

## (二) 实验题 10：求有向图的简单路径

- 题目分析

该程序包含如下函数：

PathAll1( ALGraph \* G ,int u ,int v,int path[],int d )：输出图 G 中从顶点 u 到 v 的所有简单路径。采

PathAll2( ALGraph \* G,int u,int v,int l,int path[],int d )：输出图 G 中从顶点 u 到 v 的长度为的所有

ShortPath( ALGraph \* G,int u,int v,int path[] )：求顶点 u 到顶点 v ( u ≠ v ) 的最短路径。采用从顶点

- 实验具体步骤

1. PathAll1(AdjGraph \*G,int u,int v,int path[],int d) 函数编写：

```

void PathAll1(AdjGraph *G,int u,int v,int path[],int d)
{
    ArcNode *p;
    int j,w;
    d++; path[d]=u;
    visited[u]=1;
    if (u==v && d>0)
    {
        for (j=0;j<=d;j++)
            printf("%3d",path[j]);
        printf("\n");
    }
    p=G->adjlist[u].firstarc;
    while (p!=NULL)
    {
        w=p->adjvex;
        if (visited[w]==0 )
            PathAll1(G,w,v,path,d);
        p=p->nextarc;
    }
    visited[u]=0;
}

```

2. PathAll2(AdjGraph \*G,int u,int v,int l,int path[],int d) 函数编写:

```

void PathAll2(AdjGraph *G,int u,int v,int l,int path[],int d)
{
    int w,i;
    ArcNode *p;
    visited[u]=1;
    d++; path[d]=u;
    if (u==v && d==l)
    {
        for (i=0;i<=d;i++)
            printf("%3d",path[i]);
        printf("\n");
    }
    p=G->adjlist[u].firstarc;
    while (p!=NULL)
    {
        w=p->adjvex;
        if (visited[w]==0)
            PathAll2(G,w,v,l,path,d);
        p=p->nextarc;
    }
    visited[u]=0;
}

```

3. ShortPath(AdjGraph \*G,int u,int v,int path[]) 函数编写:

```

int ShortPath(AdjGraph *G,int u,int v,int path[])
{
    struct
    {
        int vno;
        int level;
        int parent;
    } qu[MAXV];
    int front=-1,rear=-1,k,lev,i,j;
    ArcNode *p;
    visited[u]=1;
    rear++;
    qu[rear].vno=u;
    qu[rear].level=0;
    qu[rear].parent=-1;
    while (front<rear)
    {
        front++;
        k=qu[front].vno;
        lev=qu[front].level;
        if (k==v)
        {
            i=0;
            j=front;
            while (j!=-1)
            {
                path[lev-i]=qu[j].vno;
                j=qu[j].parent;
                i++;
            }
            return lev;
        }
        p=G->adjlist[k].firstarc;
        while (p!=NULL)
        {
            if (visited[p->adjvex]==0)
            {
                visited[p->adjvex]=1;
                rear++;
                qu[rear].vno=p->adjvex;
                qu[rear].level=lev+1;
                qu[rear].parent=front;
            }
            p=p->nextarc;
        }
    }
    return -1;
}

```

# 四、实验结果及分析

## (一) 实验题 5：采用普里姆算法求最小生成树

运行结果如下：

C:\ 选择 C:\Windows\system32\cmd.exe

图G的邻接矩阵：

0	5	8	7	∞	3
5	0	4	∞	∞	∞
8	4	0	5	∞	9
7	∞	5	0	5	6
∞	∞	∞	5	0	1
3	∞	9	6	1	0

普里姆算法求解结果：

- 边(0, 5)权为:3
- 边(5, 4)权为:1
- 边(0, 1)权为:5
- 边(1, 2)权为:4
- 边(4, 3)权为:5

请按任意键继续. . .

## (二) 实验题 10：求有向图的简单路径

C:\ 选择 C:\Windows\system32\cmd.exe

图G的邻接表：

- 0: 1[1]→ 3[1]→ ^
- 1: 2[1]→ ^
- 2: 0[1]→ 5[1]→ ^
- 3: 2[1]→ 5[1]→ ^
- 4: 3[1]→ ^
- 5: 0[1]→ 1[1]→ 3[1]→ 4[1]→ ^

(1) 从顶点5到2的所有路径：

- 5 0 1 2
- 5 0 3 2
- 5 1 2
- 5 3 2
- 5 4 3 2

(2) 从顶点5到2的所有长度为3路径：

- 5 0 1 2
- 5 0 3 2
- 5 4 3 2

(3) 从顶点5到2的最短路径：

- 5 1 2

请按任意键继续. . . ■

## 五、总结

(一) 通过实验 8 加深了对普里姆算法求带权通图中最小生成树的理解。

(二) 通过实验 10 进一步掌握了深度优先遍历算法和广度优先遍历算法在求解图路径搜索问题中的应用。