

图的模板类设计

林敬翊

2022 年 12 月 8 日

1 邻接表和邻接矩阵

邻接表（英语：adjacency list）是表示了图中与每一个顶点相邻的边集的集合，这里的集合指的是无序集。

如果是无向图，那么每条边由两个结点组成，分别代表边的两个端点；如果是有向图，那么每条边是一个结点对，分别代表边的始点和终点。

邻接矩阵（英语：adjacency matrix）是一种方阵，用来表示有限图。它的每个元素代表各点之间是否有边相连。

作为特例，简单图的邻接矩阵是 $(0,1)$ 矩阵并且对角线元素都为 0。无向图的邻接矩阵是对称矩阵。图和其邻接矩阵的特征值和特征向量之间的关系是谱图理论的研究对象。

图的关联矩阵需要和邻接矩阵区分。它是图的另一种矩阵表示方式，它的元素表示各个节点-边对是否相关。还有图的度数矩阵，含有每个结点的度数信息。

2 设计思路

邻接矩阵采用的是矩阵的存储方式，并且如果是有链接的两个点，将他们的双方的矩阵元素都改为 1，否则为 0。

邻接表则是将他们采用 vector 和 pushback 的方式将他们一一储存在里面。

3 测试数据与结果

```
graph g(5); //一共有多少点
g.addedge(0, 1);
g.addedge(0, 4);
g.addedge(1, 2);
g.addedge(1, 3);
g.addedge(1, 4);
g.addedge(2, 3);
g.addedge(3, 4);
g.print();
```

图 1: 邻接矩阵输入数据

0	1	0	0	1
1	0	1	1	1
0	1	0	1	0
0	1	1	0	1
1	1	0	1	0

图 2: 邻接矩阵输出结果

```

graphlist a(5);
vector<pair<int, int> > adj[5];
a.addEdge(adj, 0, 1, 10);
a.addEdge(adj, 0, 4, 20);
a.addEdge(adj, 1, 2, 30);
a.addEdge(adj, 1, 3, 40);
a.addEdge(adj, 1, 4, 50);
a.addEdge(adj, 2, 3, 60);
a.addEdge(adj, 3, 4, 70);
a.printGraph(adj, 5);

```

图 3: 邻接表输入数据

```

Node 0 ->
    Node 1 weight =10
    Node 4 weight =20

Node 1 ->
    Node 0 weight =10
    Node 2 weight =30
    Node 3 weight =40
    Node 4 weight =50

Node 2 ->
    Node 1 weight =30
    Node 3 weight =60

Node 3 ->
    Node 1 weight =40
    Node 2 weight =60
    Node 4 weight =70

Node 4 ->
    Node 0 weight =20
    Node 1 weight =50
    Node 3 weight =70

```

图 4: 邻接表输出结果