|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Ch6作业** | | | | | | | |
| 班级 | 01 | 学号 | 2022040906023 | 姓名 | 梁书恺 | 成绩 |  |
| 一、选择题（每个小题5分，共40分）  1. 无向图G有16条边，度为4的顶点有3个，度为3的顶点有4个，其余顶点的度均小于3，则图G至少有（ B ）个顶点。  A 10 B 11 C 12 D 13  2. 无向图的邻接矩阵是一个（ C ），有向图的邻接矩阵是一个（ D ）  A 上三角矩阵 B 下三角矩阵 C 对称矩阵 D 无规律矩阵  3. 具有*n*个顶点的无向图，其邻接表最多有（ B ）个边结点。  A *n*2 B *n*(*n*-1) C *n*(*n*+1) D *n*(*n*-1)/2  4. G是一个非连通无向图，共有28条边，则该图至少有（ C ）个顶点。  A．7 B．8 C．9 D．10  5. 用邻接表表示图进行广度优先遍历时，通常借助（ B ）来实现算法；深度优先遍历时，通常借助（ A ）来实现算法。  A．栈 B. 队列 C. 树 D．二叉树  6. 假设一个有向图具有*n*个顶点*e*条边，该有向图采用邻接表存储，则删除与顶点*i*相关联的所有边的时间复杂度是（ C ）。  A *O*(*n*) B *O*(*e*) C *O*(*n*+*e*) D *O*(*n*\**e*)  7. 无向图G=(V, E)，其中V ={a, b, c, d, e, f}，E = {(a, b), (a, e), (a, c),(b, e),(c, f),(f, d),(e, d)}，对该图进行深度优先遍历，得到的顶点序列正确的是（ D ）。  A abecdf B acfebd C aebcfd D aedfcb  8. 下面（ B ）方法可以判断出一个有向图是否有环。  A．深度优先遍历 B．拓扑排序 C．求最短路径 D．求关键路径  二、简答题（共40分）  1. 已知图的邻接矩阵如图6-1所示，写出从顶点*v*0出发深度优先遍历和广度优先遍历的所有可能结果。导致遍历序列不唯一的因素有哪些？    图6-1  图如下：    深度优先：0134256、0134562、0165432、0165342、0245361、0245613、0243561、0316542、0342561、0345612、0425316、0425613、0453162、0456132、0613452、0613425、0613542、0653421、0653142、0654231、0654312  广度优先：0123465…0643215  只需第一位为0，最后一位为5，其余位任意排列  不唯一因素：一些顶点邻接的顶点数大于一  2. 已知如图6-2所示的无向网，请给出：① 邻接矩阵； ② 邻接表；③ 最小生成树    图6-2    邻接矩阵：  邻接表：    最小生成树：    3. 已知无向图*G*的邻接表如图6-3所示，请画出对应的图，并写出从顶点1出发的深度遍历和广度遍历序列。  2  4  1  图6-3 无向图的邻接表存储  0  1  2  3  4  5  *v*0  *v*1  *v*2  *v*3  *v*4  *v*5  1  3 ∧  0  2 ∧  0  3  4 ∧ 5 ∧  2  4 ∧ 5 ∧    深度优先：103425  广度优先：102345  4. 有向网如图6-4所示，试用迪杰斯特拉算法求出从顶点a到其他各顶点间的最短路径，  用表格写出过程。    图6-4   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | 终点 | i=1 | i=2 | i=3 | i=4 | i=5 | i=6 | | b | 15  (a,b) | 15  (a,b) | 15  (a,b) | 15  (a,b) | 15  (a,b) | ***15***  ***(a,b)*** | | c | ***2***  ***(a,c)*** |  |  |  |  |  | | d | 12  (a,d) | 12  (a,d) | 11  (a,c,f,d) | ***11***  ***(a,c,f,d)*** |  |  | | e |  | 10  (a,c,e) | ***10***  ***(a,c,e)*** |  |  |  | | f |  | ***6***  ***(a,c,f)*** |  |  |  |  | | g |  |  | 16  (a,c,f,g) | 16  (a,c,f,g) | ***14***  ***(a,c,f,d,g)*** |  | | S | {a,c} | {a,c,f} | {a,c,f,e} | {a,c,f,e,d} | {a,c,f,e,d,g} | {a,c,f,e,d,g,b} |   三、写出算法思路（20分）  1. 判断以邻接表存储的有向图中是否存在由顶点*vi*到顶点*vj*的路径（*i*≠*j*）。  void existPath(int d, int z)  {  G->visited[d] = 1;  ArcNode \*p = G->adjlist[d].Arc;  if (d == z) connect = 1;  while (p != NULL)  {  if (p->ajdVex == z) connect = 1;  if (!G->visited[p->ajdVex]) existPath(p->ajdVex, z);  p = p->next;  }  } | | | | | | | |