数据结构部分参考答案

一、单项选择题 （每题 1 分，共 8 分）

1.（1） 2.（1） 3.（4） 4.（4）
5.（4） 6.（3） 7.（2） 8.（3）

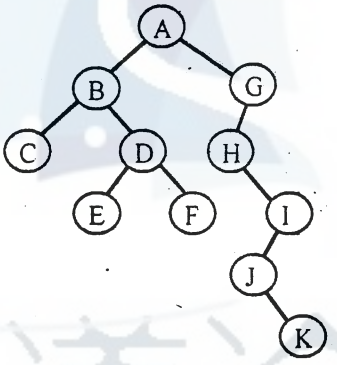
二、判断题：正确用√表示，错误用×表示。（每题 1 分， 共 5 分）

1.（×）在顺序表中取出第 i 个元素所花费的时间与 i 成正比。
2.（√）一个有向图的邻接表和逆邻接表中的结点个数一定相等。
3.（√）直接选择排序算法的时间复杂度为 $O(n^2)$，不受数据初始排列的影响。
4.（×）由于希尔排序的最后一趟与直接插入排序过程相同，因此前者一定比后者花费的时间更多。
5.（×）广义表的长度是指广义表中的原子个数。

三、简答题 （每题 5 分，共 20 分）

1. 一棵二叉树的先序、中序和后序序列如下，其中有部分未标出，试构造出该二叉树。

先序序列为：__ __CDE__GHI__K
中序序列为：CB__ __FA__JKIG
后序序列为：__EFDB__JIH__A



2. 将下面算法划线处用具体语句表示，使其完成所要求的功能。

```
PROC  ins_linklist (la ： linkisttp； i： integer； b： elemtp )；
  {la 为带头结点单链表的头指针，在 i 之前插入数据元素 b}
    p： = la ↑ .next； j： = 0；
    WHILE ( 循环控制条件 ) DO
      [ 循环体部分 ]；
    IF  (判定条件)
      THEN ERROR ( ' i 非法' )
      ELSE [ new(s)； s ↑ .data： =b；
             插入链表部分
           ]
ENDP； { ins_linklist }
```

```
WHILE (p 〈〉 nil AND j<i-1 ) DO
  [p ： = p ↑ .next； j： = j +1]；
IF  (p=nil OR j>i-1)
  THEN ERROR ( ' i 非法' )
  ELSE [ new(s)； s ↑ .data： =b；
         s ↑ .next： = p ↑ .next；
         p ↑ .next： = s  ]
```

3．对有向图和无向图，分别描述如何判定图中是否存在回路。

答： 对有向图，可以用拓扑排序算法来判定图中是否存在回路。当输出顶点数小于顶点数时，图中存在回路，否则，图中无回路。

对无向图，若边数大于等于顶点数时，则图中存在回路，否则，图中无回路。

4．试设计一种链队列的存储结构，要求只用一个指针，并且使其插入和删除的时间复杂度均为 O(1)。试画出存储结构，并指出队首和对尾。

答：采用仅有尾指针 rear 的循环单链表表示链队列（如图），队空时 rear=nil，
队首为第一个结点（fron=rear↑.next），队尾为最后一个结点（rear）。

四、算法题（共 17 分）

1．修改下面中序遍历二叉树算法，使其能判定所输出的序列是否有序。（9 分）

```
PROC inorder (bt：bitreptr);
{bt 为二叉树根结点指针}
  IF bt<>nil THEN
    [   inorder(bt↑.lchild);
        visit(bt↑.data);
        inorder(bt↑.rchild)
    ]
ENDP；{ inorder }
```

参考算法：

算法思想：用一个变量 predata 作为前趋结点的值，初值为最小值-Maxint，遍历中判断，当某结点的值小于其前趋时，输出序列为无序序列，逻辑变量 B 为 false。

```
PROC inorder (bt：bitreptr);
  {bt 为二叉树根结点指针，初次调用时 predata = -Maxint，B 为 true}
  IF bt<>nil THEN
    [   inorder(bt↑.lchild);
        IF bt↑.data < predata THEN B：=false;
        predata：= bt↑.data;
        inorder(bt↑.rchild)
    ]
ENDP：{ inorder }
```

2. 修改下面层次遍历二叉树算法，使其能判断该二叉树是否为完全二叉树。(8 分)

```
PROC level (bt: bitreptr);
  IF bt<>nil THEN
   [  INIQUEUE(Q);  ENQUEUE(Q,bt);{初始化队列 Q，并将根入队列}
      WHILE NOT EMPTY(Q) DO  {队列非空进入循环}
        [  p:=DEQUEUE(Q);{出队列}
           visit(p↑.data);
           IF  p↑.lchild<>nil THEN  ENQUEUE(Q,p↑.lchild);
           IF  p↑.rchild<>nil THEN  ENQUEUE(Q,p↑.rchild);
        ]
    ]
  ENDP; { level }
```

参考算法：

算法思想：完全二叉树中，若某结点无左孩子，也不能有右孩子；
若某结点缺少孩子，其后的所有结点就不能再有孩子。

因此，可以使用一个标志 tag 和两个计数变量 i 和 j，每访问一个结点，i 加 1，若某结点缺少孩子时，标志置为 false；计数变量 j 停止计数。

最后，当 i=j 时，为完全二叉树，否则为非完全二叉树。

```
PROC level (bt: bitreptr);
  i:=0; j:=0; tag:=true; {初始化}
  IF bt<>nil THEN
   [  INIQUEUE(Q);  ENQUEUE(Q, bt);{初始化队列 Q，并将根入队列}
      WHILE NOT EMPTY(Q) DO  {队列非空进入循环}
        [  p:=DEQUEUE(Q);{出队列}
           i:= i+1; IF tag THEN j:=j+1;
           IF  p↑.lchild<>nil THEN  ENQUEUE(Q, p↑.lchild)
                                ELSE tag:=false;
           IF  p↑.rchild<>nil THEN  ENQUEUE(Q, p↑.rchild)
                                ELSE tag:=false;
        ]
    ]
    IF  i=j  THEN  write ('是完全二叉树')
             ELSE  write ('不是完全二叉树')
ENDP; { level }
```

操作系统的参考答案

五．单项选择题：（每小题 1 分，共 10 分）

1.② 2.① 3.③ 4.① 5.① 6.② 7.③ 8.④ 9.② 10.④

六．填空题（每小题 1 分，共 9 分）

1. 512 2. 根，当前工作 3. 物理 4. 原子操作

5. bernstein 条件 6.竞争资源，进程推进顺序不当

7. 共享存储系统，消息系统，管道通信

8. 脱机作业控制，联机作业控制

9. 字节多路通道，数组选择通道，数组多路通道

七．判断题（每小题 1 分，共 10 分）

1.× 2.√ 3.× 4.× 5.√ 6.√ 7.× 8.× 9.× 10.×

八．问答题（每小题 7 分，共 21 分）

1.答：其它用户的平均等待时间：3/250

2.答：缺页次数：6 次。缺率=0.3

3.答:CPU 的平均有效访问存储器时间为:52