

4. 设栈的初始状态为空，入栈序列为 1, 2, 3, 4, 5, 6，若出栈序列为 2, 4, 3, 6, 5, 1，则操作过程中栈中元素个数最多时为 ()

- A. 2 个
B. 3 个
C. 4 个
D. 6 个

5. 队列的特点是 ()

- A. 允许在表的任何位置进行插入和删除
B. 只允许在表的一端进行插入和删除
C. 允许在表的两端进行插入和删除
D. 只允许在表的一端进行插入，在另一端进行删除

6. 一个链串的结点类型定义为

```
# define NodeSize 6
typedef struct node{
    char data[NodeSize];
    struct node*next;
}LinkStrNode;
```

如果每个字符占 1 个字节，指针占 2 个字节，该链串的存储密度为 ()

- A. 1/3
B. 1/2
C. 2/3
D. 3/4

7. 广义表 $A = (a, B, (a, B, (a, B, \dots)))$ 的长度为 ()

- A. 1
B. 2
C. 3
D. 无限值

8. 已知 10×12 的二维数组 A，按“行优先顺序”存储，每个元素占 1 个存储单元，已知 $A[1][1]$ 的存储地址为 420，则 $A[5][5]$ 的存储地址为 ()

- A. 470
B. 471
C. 472
D. 473

9. 在一棵二叉树中，度为 2 的结点数为 15，度为 1 的结点数为 3，则叶子结点数为 ()

- A. 12
B. 16
C. 18
D. 20

10. 在带权图的最短路径问题中，路径长度是指 ()

- A. 路径上的顶点数
B. 路径上的边数
C. 路径上的顶点数与边数之和
D. 路径上各边的权值之和

11. 具有 n 个顶点、 e 条边的无向图的邻接矩阵中，零元素的个数为 ()

- A. e
B. $2e$

C. n^2-2e D. n^2-1

12.要以 $O(n \log n)$ 时间复杂度进行稳定的排序, 可用的排序方法是 ()

A.归并排序

B.快速排序

C.堆排序

D.冒泡排序

13.若希望在 1000 个无序元素中尽快求得前 10 个最大元素, 应借用 ()

A.堆排序

B.快速排序

C.冒泡排序

D.归并排序

14.对有序表进行二分查找成功时, 元素比较的次数 ()

A.仅与表中元素的值有关

B.仅与表的长度和被查元素的位置有关

C.仅与被查元素的值有关

D.仅与表中元素按升序或降序排列有关

15.散列文件是一种 ()

A.顺序存取的文件

B.随机存取的文件

C.索引存取的文件

D.索引顺序存取的文件

二、填空题 (本大题共 10 小题, 每小题 2 分, 共 20 分)

请在每小题的空格中填上正确答案。错填、不填均无分。

16.若一个算法中的语句频度之和为 $T(n) = 3n^3 - 200n \log_2 n + 50n$, 则该算法的渐近时间复杂度为_____。

17.在单链表中, 除了第 1 个元素结点外, 任一结点的存储位置均由_____指示。

18.栈的修改是按_____的原则进行。

19.字符串中任意个连续的字符组成的子序列称为该串的_____。

20.假设一个 10 阶的上三角矩阵 A 按行优先顺序压缩存储在一维数组 B 中, 若矩阵中的第一个元素 a_{11} 在 B 中的存储位置 $k=0$, 则元素 a_{55} 在 B 中的存储位置 $k=$ _____。

21.在一棵具有 n 个结点的严格二叉树中, 度为 1 的结点个数为_____。

22.对于稀疏图, 采用_____表示法较为节省存储空间。

23.在排序过程中, 如果_____, 则称其为外部排序。

24.设有一组记录的关键字为 {19, 14, 23, 1, 68, 12, 10, 78, 25}, 用链地址法构造散列表, 散列函数为 $h(\text{key}) = \text{key} \% 11$, 散列地址为 1 的链中有_____个记录。

25.多关键字文件的特点是除主文件和主索引外, 还建有_____。

三、解答题 (本大题共 4 小题, 每小题 5 分, 共 20 分)

26.对于下列稀疏矩阵 (注: 矩阵元素的行列下标均从 1 开始)

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 7 & -1 & 0 & 0 \\ -8 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6 & -2 & 9 \end{bmatrix}$$

- (1) 画出三元组表;
- (2) 画出三元组表的行表。

- (1)
- (2)

27.已知一个森林的前序遍历序列为 CBADHEGF, 后序遍历序列为 ABCDEFGH。

- (1) 画出该森林;
- (2) 画出该森林所对应的二叉树。

- (1)
- (2)

28.对关键字序列 (429, 653, 275, 897, 170, 908, 473, 256, 726) 进行基数排序, 写出每一趟的排序结果。

29.对下列关键字序列

(87, 25, 310, 08, 27, 132, 68, 96, 187, 133, 70, 63, 47, 135)

构造散列表, 假设散列函数为 $h(\text{key}) = \text{key} \% 13$, 用拉链法解决冲突。

- (1) 画出该散列表;
- (2) 求等概率情况下查找成功的平均查找长度 ASL;
- (3) 写出删除值为 70 的关键字时所需进行的关键字比较次数。

- (1)
- (2)
- (3)

四、算法阅读题 (本大题共 4 小题, 每小题 5 分, 共 20 分)

30.阅读下列算法, 并回答问题:

- (1) 假设 $L = (3, 7, 7, 11, 20, 20, 20, 51, 51)$, 写出执行函数 $f30(\&L)$ 后的 L ;
- (2) 简述 $f30$ 的功能。

```
void f30(SeqList*L)
{ //L 为非空的有序表
  int i=1,k=0;
  while(i<L->length)
  {
```

```
if(L->data[i]!=L->data[k])
L->data[++k]=L->data[i];
i++;
}
L->length=k+1;
}
(1)
(2)
```

31.阅读下列算法，并回答问题：

- (1) 假设栈 S= (3, 8, 6, 2, 5)，其中 5 为栈顶元素，写出执行函数 f31 (&S) 后的 S；
- (2) 简述函数 f31 的功能。

```
void f31(Stack *S){
    Queue Q; InitQueue(&Q);
    while(!StackEmpty(S))
        EnQueue(&Q,Pop(&S));
    while(!QueueEmpty(Q))
        Push(&S,DeQueue(&Q));
}
```

- (1)
- (2)

32.假设具有 n 个结点的完全二叉树顺序存储在向量 BT[1.. n]中，阅读下列算法，并回答问题：

- (1) 若向量 BT 为：

A	B	C	D	E	F	G
1	2	3	4	5	6	7

画出执行函数 f32 (BT,7,1) 的返回结果；

- (2)简述函数 f32 的功能。

```
BinTree f32(DataType BT[],int n,int i)
{
    BinTree p;
    if (i>n) return NULL;
    p= (BinTNode*) malloc(sizeof(BinTNode));
    p->data=BT[i];
```

```
p->lchild=f32(BT,n,i*2);  
p->rchild=f32(BT,n,i*2+1);  
return p;  
}
```

(1)

(2)

33.已知有向图的邻接表和邻接矩阵定义如下:

```
# define  MaxNum  50                                // 图的最大顶点数  
typedef  struct  node {  
    int  adjvex;                                     // 邻接点域  
    struct  node  *next;                             // 链指针域  
}  EdgeNode;                                        // 边表结点结构  
typedef  struct {  
    char  vertex;                                    // 顶点域  
    EdgeNode  *firstedge;                            // 边表头指针  
}  VertexNode;                                     // 顶点表结点结构  
typedef  struct {  
    VertexNode  adjlist [MaxNum];                   // 邻接表  
    int  n,e;                                        // 图中当前顶点数和边数  
}  ALGraph;                                        // 邻接表描述的图  
typedef  struct {  
    char  vertex[MaxNum];                            // 顶点表  
    int  adjmatrix [MaxNum][MaxNum];                // 邻接矩阵  
    int  n,e;                                        // 图中当前顶点数和边数  
}  AMGraph;                                        // 邻接矩阵描述的图
```

下列算法是将邻接表描述的图 G1 改为邻接矩阵描述的图 G2, 在空白处填上适当内容使算法完整:

```
void  f33 (ALGraph G1,AMGraph *G2)  
{  int  i, j;  
    EdgeNode  *p;  
    G2->n=G1.n;  
    G2->e=____(1)____;  
    for  (i=0; i<G1.n; i++)  
    {  G2->vertex[i]=____(2)____;
```

```
p=G1.adjlist[i].firstedge;
for (j=0; j<G1.n; j++) G2->adjmatrix[i][j]=0;
while (p)
{   G2->adjmatrix[i][p->adjvex]=1;
    _____(3)_____;
}
}
```

- (1)
- (2)
- (3)

五、算法设计题（本题 10 分）

34. 设顺序表 L 是一个递增有序表。编写算法，要求利用二分查找法确定插入位置，将元素 x 插入到 L 中，使 L 保持有序。