

习题 2 表结构

申昱 22920192204266

2-1 B

2-2 C

2-3

(1) 从时间角度看，二维数组在插入数据和读取数据时时间复杂度为 $O(1)$ 。做加法时可以直接对应进行数据的相加，需遍历一次数组，时间复杂度为 $O(n^2)$ 。

三元组顺序表做加法时可以构造十字链表，时间复杂度为 $O(A \text{ 矩阵非零元素个数} + B \text{ 矩阵非零元素个数})$ 。

(2) 从空间角度看，二维数组需要开辟 n^2 空间，对空间的浪费较大。三元组顺序表：只需要开辟非零元素数量的空间，对稀疏矩阵来说更节省空间。

2-4

(1) 顺序表在内存中是一段连续的存储空间，可以通过头指针和偏移地址直接访问数据，访问数据的效率较高，为 $O(1)$ 但是插入数据和删除数据的效率较低。

(2) 链表在内存中是不连续的，通过每一个节点保存指向下一个节点的指针的方式来存储数据，访问数据的效率较低，但同时，插入数据和删除数据的效率较高。

2-5

头指针：指向头节点的指针。

头节点：一般为一个空节点，next 指针指向首元素节点。

首元素节点：链表中第一个存储着数据的节点。

2-6 B

2-7 A

2-8 A

2-9 A

2-10 123 132 231 213 321

2-11 无法得到 435612，因为此时 2 应在 1 之前出栈。可以得到 135426。

2-12 将队列 Q 中的元素进行翻转。

2-13 寻找 m 和 n 的最大公约数。

2-14

```
int g(int m, int n)
```

```
{
```

```
    if(m==0 && n>=0)
```

```
        return 0;
```

```
    else
```

```
        return g(m-1.2*n)+n;
```

```
}
```

2-15 void test1(int &s)

```
{
```

```
    int x;
```

```
    cin>>x;
```

```
    s = 0;
```

```
    while(x!=0)
```

```
    {
```

```
        s += x;
```

```
        cin >> x;
```

}
}

2-16 1 $A \rightarrow C$

2 $A \rightarrow B$

1 $C \rightarrow B$

3 $A \rightarrow C$

1 $B \rightarrow A$

2 $B \rightarrow C$

1 $A \rightarrow C$