## **第三章 栈和队列**

1. 设依次进入一个栈的元素序列为c,a,b,d,不可得到出栈的元素序列有\_\_\_ A \_\_。

A.a.b,c,d B.a,d,c,b C.b,a,d,c D.c,d,a,b

2. \_\_ C \_\_又称为FIFO表。

A.队列 B.散列表 C.栈 D.哈希表

## 3．栈是一种 A 的线性表。

A. 只允许在一端进行插入和在另一端进行删除

B. 只允许在一端进行插入和删除

C. 只允许在两端进行插入和删除

D. 允许在中间部位进行插入和删除

4、循环队列用数组A[0，m-1]存放其元素值，已知其头尾指针分别是front和rear，则当前队列中的元素个数是\_\_\_ A \_。

A．(rear-front+m)%m B．read-front+1

C．read-front-1 D．read -front

5、写出下列程序段的运行结果（栈中的元素类型是char）：

main(　)

{ SEQSTACK s,\*p;

char x, y;

p = &s;

initstack(p);

x = ‘c’; y = ‘k’;

push(p,x); push(p,’a′); push(p,y);

x = pop(p);

push(p,’t′);

push(p,’s′);

while(!empty(p))

{ y = pop(p);

printf(″% c″,y);}

printf(″%c＼n″,x);

}

## 运行结果是 stack 。

6、设Q为顺序队列，InitQueue函数为初始化队列，EnQueue函数为进队，DelQueue函数为出队，QueueEmpty函数为判队列是否为空。读下列算法，写出输出结果。

void func1(Queue Q)

{ InitQueue(Q);

int a[5]={10,20,30};

for (i=0;i<3;i++) EnQueue(Q,a[i]);

EnQueue(Q,DelQueue(Q));

EnQueue(Q,62);

EnQueue(Q,DelQueue(Q)+10);

while (!QueueEmpty(Q)) printf(“%3d”,DelQueue(Q));

}

30 10 62 30

7、写出用p(5)调用以下递归函数的输出结果。

void p(int n)

{int k;

if(n)

{p(n-1);

for(k=n;k>=1;k--) printf(“%d”,n);

printf(“\n”);

}

}

**1**

**21**

**321**

**4321**

**54321**