

1.已知二叉树的先序访问序列 ABDGECF，中序访问序列 DGBEAF C，请写出后续访问序列。

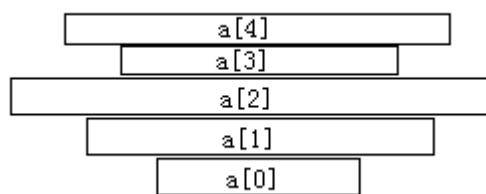
2.设顺序循环队列中，队头指示器 **front**，队尾指示器 **rear**，存储单元个数 **maxsize**。写出入队和出队时 **rear** 和 **front** 的计算公式，队空时的判断公式，队满时的判断公式，队列长度计算公式。

3、已知单链表中结点类为 **Node**，**p** 指向第 **i-1** 个结点，若将第 **i** 个结点删除，所使用语句包括：**p->next=q->next; x=q->data; Node<T> *q=p->next; delete q;** 请给出这些语句的正确排列顺序。

4、给定一组权值{6， 2， 3， 3， 4， 9}用哈夫曼算法构造哈夫曼树。

编程题：

1、如图表示大小不一的一摞饼，**a[i]**表示第 **i** 层饼大小。设计一个类 **F** 能对 **n** 个饼进行翻转，使得大饼在下小饼在上排放。要求每次只能选择上方的一堆饼，一起翻转。



答案:

1.答案: GDEBFCA (4分)

2.答案:

出队时: $\text{front}=(\text{front}+1)\% \text{maxsize}$ 入队时: $\text{rear}=(\text{rear}+1)\% \text{maxsize}$

队空时: $\text{front}==\text{rear}$ 队满时: $(\text{rear}+1)\% \text{maxsize}==\text{front}$

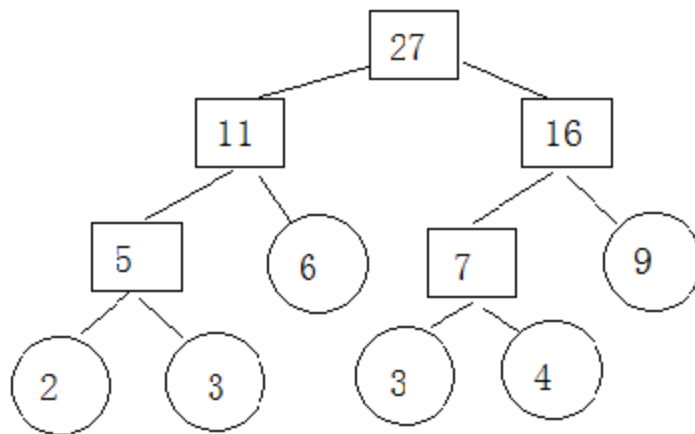
队列长度: $(\text{maxsize}+\text{rear}-\text{front})\% \text{maxsize}$

(错一个扣1分。)

3.答案: `Node<T> *q=p->next; x=q->data; p->next=q->next; delete q;`

或 `Node<T> *q=p->next; p->next=q->next; x=q->data; delete q;`

4.答案:



编程题:

1. 答案:

```
#include <iostream>
```

```
using namespace std;
```

//头文件和命名空间2分

```
class F{
```

```
private:
```

```
    int *a;int n;
```

```
    void sort(int *a,int n);
```

```
    void swap(int *low,int *high);
```

```
public:
```

```
    F(int *a1,int n1):a(a1),n(n1){}
```

```
    void sort1(){sort(a,n);}
```

```
};
```

```
void F::sort(int *a,int n)
```

```
{
```

```
    if(n==1) return;
```

```
    int max_i=0;
```

```
    int max_val=*a;
```

```
    for(int i=0;i<n;i++)
```

```
    if(a[i]>max_val){max_val=a[i];max_i=i;}
```

```
    swap(&a[max_i],a+n-1);
```

```
    swap(a,a+n-1);
```

```

sort(a+1, n-1);
}
void F::swap(int *low, int *high)
{
    int temp;
    while(low<high)
    {
        temp=*low;
        *low=*high;
        *high=temp;
        low++;
        high--;
    }
}
Void main() {
int a[]={2, 5, 8, 4, 12, 63, 18, 73, 52, 31};
    F b(a, 10);
    b.sort1();
    for(int i=0; i<10; i++) cout<<a[i]<<" ";
}

```