

```

typedef int Position;
struct QNode {
    ElementType *Data;      /* 存储元素的数组 */
    Position Front, Rear;   /* 队列的头、尾指针 */
    int MaxSize;           /* 队列最大容量 */
};
typedef struct QNode *Queue;

Queue CreateQueue( int MaxSize )
{
    Queue Q = (Queue)malloc(sizeof(struct QNode));
    Q->Data = (ElementType *)malloc(MaxSize * sizeof(ElementType));
    Q->Front = Q->Rear = 0;
    Q->MaxSize = MaxSize;
    return Q;
}

bool IsFull( Queue Q )
{
    return ((Q->Rear+1)%Q->MaxSize == Q->Front);
}

bool AddQ( Queue Q, ElementType X )
{
    if ( IsFull(Q) ) {
        printf("队列满");
        return false;
    }
    else {
        Q->Rear = (Q->Rear+1)%Q->MaxSize;
        Q->Data[Q->Rear] = X;
        return true;
    }
}

bool IsEmpty( Queue Q )
{
    return (Q->Front == Q->Rear);
}

ElementType DeleteQ( Queue Q )
{
    if ( IsEmpty(Q) ) {
        printf("队列空");
        return ERROR;
    }
    else {
        Q->Front = (Q->Front+1)%Q->MaxSize;
        return Q->Data[Q->Front];
    }
}

```