|  |
| --- |
| #define MAX 5    typedef struct queue  {      int front   ;      int rear    ;      int ele[MAX]    ;  }Queue;    void init(Queue \*q)  {      q->rear  = -1;      q->front =  0;  }    int isFull(Queue \*q)  {      int full=0;        if( q->rear == MAX -1)          full = 1;        return full;  }    int isEmpty(Queue \*q)  {      int empty=0;        if( q->front == q->rear+1 )          empty = 1;        return empty;  }    void insertQueue(Queue \*q,int item)  {      if( isFull(q) )      {          printf("\nQueue Overflow");          return;      }        q->ele[++(q->rear)] = item;      printf("\nInserted item : %d",item);  }    int deleteQueue(Queue \*q, int \* item)  {      if( isEmpty(q) )      {          printf("\nQueue Underflow");          return -1;      }      \*item = q->ele[(q->front)++];      return 0;  }    int main()  {      int item = 0;      Queue q;        init(&q);        insertQueue(&q,10);      insertQueue(&q,20);      insertQueue(&q,30);      insertQueue(&q,40);      insertQueue(&q,50);      insertQueue(&q,60);        if( deleteQueue( &q, &item ) == 0 )          printf("\nDeleted item : %d",item);      if( deleteQueue( &q, &item ) == 0 )          printf("\nDeleted item : %d",item);      if( deleteQueue( &q, &item ) == 0 )          printf("\nDeleted item : %d",item);      if( deleteQueue( &q, &item ) == 0 )          printf("\nDeleted item : %d",item);      if( deleteQueue( &q, &item ) == 0 )          printf("\nDeleted item : %d",item);      if( deleteQueue( &q, &item ) == 0 )          printf("\nDeleted item : %d",item);        printf("\n");      return 0;  } |