```
CIRCULAR QUEUE
#inclode < stolo. hs
#include astdlib. hs
#include = process.h>
#deline que-size 3
 ant atem, front = 0, rear = -1, or (que-size), count = 0;
  void inset real ()
  f it (coort = = que - size)
   Prot (" queue overflow");
       whom;
   real = (real +1) . !. que sige;
      of Creat = item;
      count +4;
   int delete front ()
   2 if (court = = 0) return 4;
      item = q (front);
     front = (front+1).1. que - size;
       count = count -1;
      return etern;
      word display ()
   2 ent 3, f;
```

```
"if ( count = = 0)
  & prott (" queue & empty");
    return;
   fofront:
   Print ("contents of queue (n");
    for (1=0; 1 = count; 1++)
    print (0.1.d (n 0, q(+));
     f=(++1).1. que - 8ize;
   4
4
  void main ()
    for (;;)
Printf ("In1: Insect read In2: Delete front In3: Display In4: exit h
    Printf (" Gree the choice: ");
     Sconf (". (. d! & choice);
    switch (chaice)
  case 1: prints ("enter 9 tem to be insuled: ");
           scanf (". lod", & "ten);
           insert real ();
           break;
 case 2: item = deleterrant ().
     if (9ten = = -1)
   Print ( "queue is empty In ");
     else
    Printf (" ten deleted is . I.d (n", item).
       break;
  Cax 3: displaya();
```

```
break ();
                              · ( ti) in ," of tole" I leading
 default: exit(0);
4
  LINEAR QUEUE
 Finclode estato-hs
 Hinclude L Stdlebohs
 # define QUE_SIZE 3
 ent etem, front =0, real =1, q, Clas
   void insertreal ()
  & if (real = = QUE_SIZE -1)
    ¿ print ("quoue oreetlow | n");
      return ; had son at wall at 060 3 has
      real = real +1;
      V [real] = : ben;
    & if (front smeal)
      front so; ("al pages of every") these
       return -1 mote " of boto a bitalib or of progr
     return of [front ++]
     Vord displaya ()
   it (front sreal)
   ¿ pratt ("queue is empty (n");
      Print (" contents of queue ( ");
        for (1 = front: ic=real; 9++)
```

```
print (".1.2 (n", v(3));
 int main ()
int choice;
   tor(;;)
  3
Printf ("1: insect real In 2: deleté tront (n3: displayin4: exit p
    printf (" Enter the choice (n");
     scant (". 1.d", & choice);
     switch (choice)
    case! : prentf (" Enter the item to be insected (n");
             scart (0. (0d 0, 48tem);
             insut real ();
              break;
   Case 2: :tem = delete front ();
             if (item = = -1)
           print ("queue is empty (n");
           else
            print (" :tern deleted = · ( d (n", item);
            break;
    Case 3: displaya()
              break ;
          default: exit(0);
      4
```