LAB-4;
: Double Ended Queue:
(as(1)) + 2
#include < Stdio.h>
include < Conjoh?
define asize 5 1 mil 1
(/ / int of = 0, x = +1, en;) / fried
int item, q [10];
int isfull()
10+ 13 ruil () (4) (++ q 101921) 2)
return (== qsize+1)? 1:0; 3
Committee of the contraction of
int isempty ()
[[-(0)] { return (f>r)? 1:00; 3 4:00
void insert rear ()
¿ if (isful()) &
printf ("Queue overflow/n").
return's 3-17-379878 Liov
8=8+1 (0=1+1=1)
(1/2) + 9 (r) z+jem; 13 (1-7)= 1:3:
void delete front ()
2
if (isempty ())
& printf ("que ue empty m");
Marchan in in metuen in 30
printf ("item deleted is 1-d \n")
Q[(+)++]);
14 (+ 22)
2 (Chitamos)
Marine 2 21 Supring "D 2 Tricke 3.
La de la constanta de la const

```
· void insert_tront()
  3 if (f120)
      & f=f-1; q[f]= item; retuen;}
      else if ((f=0) 44 (r==-1))
         3 q[++(r)]= jtem, return; }
       else
            printf ( "inserteon not possible \n");
 void delete_rear ()
     if (isempty ())
          print f ("Queue is empty \n");
return;
   print ("item deleted is 1.d ln", q[(r)--]);
if (f>r)
        f=0; r==1; 3 +000 6
 void insert_front()
   & if (f!=0)
       2. f = f-1; 9 [f] = item; return; ]
      else if ((f==0) NA (v==-1))
            etexa) Jeilen; returnige
      ector ide
           { q[++(r)] = item; return; }
     else
             printf ("insertion not possiblela)
Void delete_rear ()
     if (isempty())

§ print f ("Queue is empty \n").
              return; ?
```

printf ("Item deleted is Itelm") If (f>r) If (f end in the item in in in item
void main() 2 for(;;) 2 printf ("1. insert_rear from 1 m 3. delete_rear m 3. delete_rear m 6. exit m"); Scanf ("1.d", pich); Switch (ch) 2 care 1: printf ("enter the item m"); Scanf ("1.d", pitem); Insert_rear();
void main() 2 for(;;) 2 printf ("1. insert_rear from 1 m 3. delete_rear m 3. delete_rear m 6. exit m"); Scanf ("1.d", pich); Switch (ch) 2 care 1: printf ("enter the item m"); Scanf ("1.d", pitem); Insert_rear();
void main() 2 for(;;) 2 printf ("1. insert_rear from 1) 3. delete_rear m 3. delete_rear m 6. exit m"); Scanf ("1.d", pich); Care 1: printf ("enter the item m"); Scanf ("1.d", pitem); Insert rear();
void main() 2 for(;;) 2 printf ("1. insert_rear from 1) 3. delete_rear m 3. delete_rear m 6. exit m"); Scanf ("1.d", pich); Care 1: printf ("enter the item m"); Scanf ("1.d", pitem); Insert rear();
void main() 2 for(;;) 2 printf ("1. insert_rear from 1) 3. delete_rear m 3. delete_rear m 6. exit m"); Scanf ("1.d", pich); Care 1: printf ("enter the item m"); Scanf ("1.d", pitem); Insert rear();
for (;i) 2 print f (" 1. insert in ear In 2. insert front In 3. delete rear In 5. display In 6. exit [n"]; Scanf (" 'Id", pich); Switch (ch) 2 care 1: print f ("enter the item In"); Scanf ("'I.d", pitem); insert rear();
for (;i) 2 print f ("1. insert rear from 1) 3. delete rear m 3. delete from 1 m 5. display m 6. exit (n"); Scanf ("1.d", pich); Switch (ch) 2 care 1: print f ("enter the item m"); Scanf ("1.d", pitem); Scanf ("1.d", pitem); Insert rear();
2 printf (" 1. insert_rear m. 2. insert_front m 3. delete_rear m 4. delete_front m 5. display m 6. exit [m"); Scanf (" 1d", pich); Switch (ch) 2 care 1: printf ("enter the item m"); Scanf ("1.d", pitem); Insert_rear();
3. delete_rear m 3. delete_rear m 4. delete_front m 5. display m 6. exit \ n"); Scanf (" of one choice m"); Switch (ch) 3. delete_front m 5. display m 6. exit \ n"); Scanf (" one choice m"); Switch (ch) 3. delete_front m 6. exit \ n"); Scanf (" one choice m"); Scanf (" one choice m"); Scanf (" one choice m"); Insert rear();
3. delete_rear m 3. delete_rear m 4. delete_front m 5. display m 6. exit \ n"); Scanf (" 1.d", pich); Switch (ch) 3 Cane 1: printf ("enter the item m"); scanf ("1.d", pitem); insert rear();
3. delete_rear m 4. delete_front m 5. display m 6. exit (n"); Scanf (" 1d", pich); Switch (ch) 2 Carel: printf ("enter the item m"); Scanf ("1.d", pitem); Insert rear();
print f (" enter choice ha"); Scanf (" 1d", pich); Switch (ch) Care 1: print f ("enter the item ha"); Scanf ("1.d", pitem); insert rear();
print f (" enter choice m"); Scamf (" 1d", pich); Switch (ch) Canel: print f ("enter the item/n"); Scamf ("1.d", pitem); insert rear();
print f (" enter choice lm"); Scamf (" 1.d", pich); Switch (ch) Example ("enter the item!n"); Scamf ("1.d", pitem); insert rearcy;
print f (" enter choice lm"); Scamf (" 1.d", pich); Switch (ch) Example ("enter the item!n"); Scamf ("1.d", pitem); insert rearch;
Switch (ch) 2 Cane 1: printf ("enter the item/n"); scamf ("1.d", pitem); insert rearch;
Switch (ch) 2 Cane 1: printf ("enter the item/n"); scamf ("1.d", pitem); insert rearc);
Switch (ch) 2 Cane 1: printf ("enter the item/n"); scamf ("1.d", pitem); insert rearch;
care!: printf ("enter the item/n"); scaref ("1.d", pitem); insert rearc);
scamf ("1.d, pitem), insert rearch;
scamf ("1.d, pitem), insert rearch;
inseit rearching
break; male = [noort]
Better the state of the state o
cases: printf ("enter the item m");
Scanf ("1.d", & item);
in seit front () jaloto bill
break;
Case 3: delete_rear ();
break; 77 = mit
case 4; delete-front();
break; +min
Case 5; display ();
break;

```
default: exit(0);
Practice: Circular Queue; Circular Aueue;
# include < stdio.h>
# include < conjo. h > mil. 1 ) flaren
 # define Que size 3
 int item, front 20; rear = -1, q[Que_Size],
  void insertrear()
   pf ("queue Overflow \n");

return; island thereof
   rear = (rearti).1. QUE_SIZE;
 q [rear] = item; [rear] = (3) (4) (4) (4) (5) (5) (5)
  Int delete front ()
    if (count = = 0) return -1;
  item = q [front];
front = (front +1) 1. QUE_SIZE;
      Count = Count-1;
      return item;
```

```
void display Q () ( ) === ( ) }
2 int i,f;

if (count = =0)

2 pf ("queue is empty m");

return;
  f = front;

printf ("Contents of Quene m");

for (iz1; i<=Count; i++);
       printf (".1.d/n", q[f];

f = (f+1).1. QUE_SIZE;
    3
 Void main ()
      int choice;
       prints (" In1: insert rear
                       m2: delete front
          printf ("enter the Choice \n");

Scanf ("1.2, & Choice);
     switch (choice)
           case!: printy ("enter items to be
inserted in");
Scanf ("1.d", &item);
                       insert rear ();
                        break;
```

