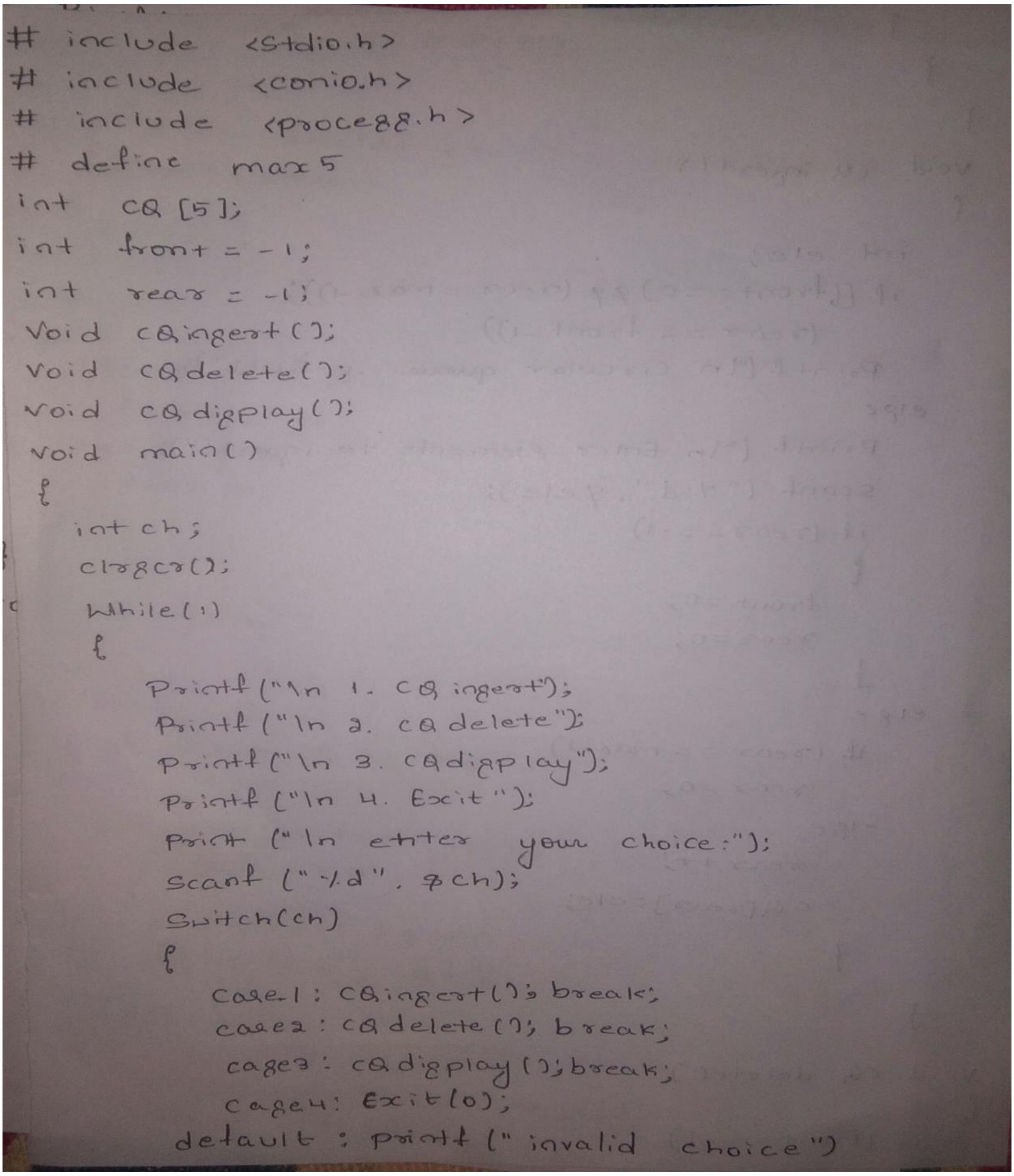
Data Structures and Algorithms

QUESTION 1 :

Write a program implementing insert, delete and display operation of Circular Queue



```
# include   <Stdio.h>
# include   <conio.h>
# include   <process.h>
# define   max5
int   CQ [5];
int   front = -1;
int   rear = -1;
Void  CQinsert ();
Void  CQdelete();
Void  CQ display();
Void  main()
 {
   int ch;
   clrscr();
   While(1)
   {
      Printf ("\n 1. CQ insert");
      Printf ("\n 2. CQ delete");
      Printf ("\n 3. CQdisplay");
      Printf ("\n 4. Exit ");
      Print ("\n enter  your  choice:");
      Scanf (" %d", &ch);
      Switch(ch)
      {
         Case1 : CQinsert(); break;
         case2 : CQ delete (); break;
          case3 : CQ display ();break;
          case4: Exit(0);
      default : print ("  invalid  choice")
```

```
        }
     }
  }
  void  cQ insert()
  {
       int ele;
       if ((front==0) && (rear = max-1));
            (rear == front -1))
        Printf("\n Circular queue is FULL");
     else
        Printf ("\n Enter Element to insert:");
        scanf ("%d", &ele);
        if (rear==-1)
        {
            front =0;
            rear=0;
        }
     else
        if (rear == max-1)
            rear =0;
        else
            rear ++;
            cQ[rear]=ele;
        }

     }
 void cQ delete()
 {
     if (front ==-1)
 Printf ("\n CQ is EMPTY");
else
 {
   Printf ("\n deleted Element = %d; CQ[front]")
   if (front = = rear)
   {
     front = -1;
     rear = -1;
   }
 else
   if (front = = Max-1)
       front = 0;
     else
       fronot ++;
   }
 }
 void CQ display()
 {
    int i;
    if (front = = -1)
        Printf ("\n CQ is EMPTY");
      else
       {
         Print ("\n front ->");
         if (front <= rear)
         {
           for (i = front; i ≤ rear; i++)
           Printf ("%d", CQ [i]);
         }
      else
         {
          for (i = front; i <= max -1 ; i++)
            Printf ("%d", CQ [i]);
        for(i=0 ; i<=rear ; i++)
            printf(" +.d ", cQ [i]);
        }
      Printf ("rear");
    }
  }
```

O/P:

```
1. cQ ingert.
2. cQ delete.
3. cQ digplay
4. Exit
    Enter your choic : 1
    Enter Element to insert : 20.

1. cQ insert.
2. cQ delete
3. cQ digplay.
4. Exit.
    Enter your choice : 1
    Enter Element to insert : 35

1. cQ ingert.
2. cQ delete
3. cQ digplay.
4. Exit
    Enter your choice : 1
    Enter element to ingert : 45

1. cQ ingert
2. cQ delete
3. cQ digplay.
4. Exit
    Enter your choice : 3
    front -> 20 35 45 rear
```

```
1. CQinsert
2. CQdelete
3. CQ display
4. Excit
   Enter your choice :2
   deleted element :20


 1. CQinsert
 2. CQdelete
 3. CQdisplay
 4. Exit
    Enter your choice :3
    front → 35    45    rear

  1. CQinsert
  2. CQdelete
  3. CQ display
  4. Exit
     Enter your choice :4.
```