```
weether and the winter
(1)
                 #includexstdio.h>
                                                                                                                              at the transport of the
                # include (stallb.h>
                                                                                                                                THE THE PART HE STATE OF THE ST
                # Enclude < malloc. h>
                                                                                                                                             and that a story thing
                    Struct node &
                                                                                                                                                  Completed of the line
                          int value;
                       Struct node * next;
                   3;
                   void insert();
                                                                                                                                                                                                 11 July 1
                   void display();
                   void delete();
                                                                                                                                                                                                            10001
                  int count():
                  typeder structorede DATA_NODE;
                 DATA_NODE * head_mode, * first_node, *temp_mode = 0, *prev_node,
                  * next_node;
                  int data;
                                                                                                                                                                                               there's want
                  int main()
                    int option=0:
                  printf ("singly Linked List Example - Alloperationsln");
                  while (option < 5)
                       prentf("In OptionsIn"):
```

```
printf ("1: Insort Proto Lenked (Ext In"):
 printf('2: Delete from LEnked KESTID'):
 printif ("3: Display Lenked LESK In"):
                                                                                                                                                                                          Late Chenge
  prints (" 4: Count Linked List In");
  printif ( Others: Exit()10"):
                                                                                                                                                                                                              1111111
     printf ("Enter your option:"):
    scanf ("1.d", soption);
    switch (option)
       case 1: Previt();
                               breakes;
        case 2: delete():
                                         break:
         case 3: display();
                                                                                                                              the first discount that
                                             break:
       case 4: count()
                                              break:
     Caxe 5
                                                                                                                                                                                                                      ; NAME OF
         defaut : breat :
          4
     4
                                                   the mental paller and the state of the state
               return o;
       4
 void enwite)
printf("In Enter Element for Insert Lenked List: In");
```

```
Scanf ('1.d', sdata);
temp_node = (DATA_NODE *) malloc(SizeOf (DATA_NODE));
· temp_node -> ratue = data:
                          Diposite & Project Strains
87 (first-rade = =0)
   first_node = temp_node;
  4
 elle
   head-oode -> next = temp-node;
 3
  temp_node > next = 0;
  head_node = temp_rode:
  ffush (staln):
                                      5 31 . 1 1
 4
                      tuly map on the state of the
 void delete()
  Pot countralue, pos, 9=0;
  count value = count();
   temp_mode = first_model):
  prentf("In Display Leoked Rist: In");
  print ("Enter posetion for Delete Element: In"):
  scarf ('1.d', & pos);
  Pf (pas >0 & & pas < = count value)
    9f (pos==1)
```

```
temp_mode = temp_mode -> next;
 print f('In Deleted successfully Inlo");
y else
while (temp_mode 1=0)
  9p (9== (p08-1))
                    the grid by a second
   prev_oode->next = temp-node->next;
   of ( P==(count value-1))
     head_node
 printf("Deleted successfullyInIn");
 break:
4
elle
 1++;
prek-mode = temp-mode:
 temp_mode = temp_mode -> next;
                               Scanned with CamScanner
```

```
part + ( 'In Invalled position (1/10)
 4
void display ()
8
   int count = 0;
   temp_oode = frrit_node;
   print-f("Display kenked kist: 10"):
   while (temp_mode!=0)
    print f ("# " od #", temp_node -> value):
  9
    count ++;
    temp-orde = temp-mode ->next;
   4
  printf ("In No of stems en Leoked List: "odln", count):
 3
                               FULL FIRE
  ent count()
4
   int count = 0;
   temp-node=first_node;
    conste (temp-orde 1 = 0)
                          house being and
   4
      count +t;
     temp-sode = temp-sode -> next;
      print + ("In No of Plems in Linked Hist: 1.dln", count);
      return count;
                                 Scanned with CamScanner
```

```
# Frodude < std 80. h >
                               Edulate C. Lange A. J. Hall
(2)
   # include (stdiets)>
   Struct Node
      ent data ;
     struct Node * next
   void prenthist (struct node * head) head)
    8
        Struct Mode * ptr = head
        whele (ptr)
          prestf (" lod -> ", ptr->data);
                             OTHER WATER TO STATE TO STATE OF THE STATE OF
       printf(" NOLLIN");
   voed push Cetruct Mode ** head, ent data)
      struct Mode * new Mode = (Struct Mode *) mallor (size of fetruct
      new Made -> data = data
      new Mode -> next = * head:
       * head = rewhode;
                          the same part of the
     4
            Historial board of the part of the part
```

```
Struct Node & shuffle Merge (Struct Node * a, struct Node * b)
3
   Struct Node during:
  strud Node * tall = slummy:
   dummy . mext = NULL:
 whale(1)
    Pf (a == NULL)
                          or hallen's the
       tall -> next = b:
     y break:
                                    The latest the state of
     else of (b == NULL)
        tall-next-as
        break:
                             Charles Telling
     4
    elle
       tall-next = a:
         tall = a;
         a = a -> next;
        tact - next = b;
         tall = b;
        b=b-next;
    return dummy next:
```

```
and main (void)
                                                                                                                             The state of the s
                  ent keeps [] = {1,2,3,4,5,6,7};
                     int n = 8132 of (keyk) / 8132 of (keys [O]);
                  Struct Node *a= NULL, *b= NULL;
                for (not 8=1) + 1 = 0; 9=1-2)
                                    push (&a, keys[1]);
           priot f ("First List: "):
            print hist (a);
        prent- ("second hist: ");
       prentlin(b);
   Struct Mode * head = Shuffle Merge (a, b);
                                                                                                                                                                                                                                      the of
      printf ("Often Menge");
    print Liet (head);
       return 0;
```

```
(3)
  #Enclude Estato.hs
   Pot top=-1;
   Pot x:
   Chan stack[100];
   void purp (9nt 1):
    Char pop ();
    ent malp()
    f
     int i,n,a,t,k,f, eum=0, eount=1;
    print ('Enter no of elemente en etack');
                             the the street of the best will
     seart ("1.d", 80)
    tor(1=0;1(0;1++)
                                     (a him has been
     forint-f( "Enter next element"):
     scarf ("1.d", fa);
     push (a);
     3
     prentf ( 'enter sum to be checked ");
    scarf (" 1.d", sk);
    for(1=0;9<0;9+t)
                                        MALL
                              I THE PROPERTY
        t=pop(1;
       sum+=t;
       count+=1
```

```
1f(sum = = k)
  for (PM j=0; ) ( count ; j++)
  printf ("1.d", stack[]]);
 f=1:
  break :
 purb(t)
                  Fire a part for the
 st (41=1)
 prints ("The Eliments in stack do not add up to the sum");
void push (int x)
 14 (top== 99)
  f printf ('In stock is full !!! In");
     return;
    top=top+1;
y stack[top]= 1:
char popes
  3+ (stack[top] == -1)
     printf ("In stack &s EMPTY!!!lo");
```

```
return o:
       x=stack[top];
         top = top-13
         return 2:
(4)
    # & Dolude < std10. h>
    # define SIZE 10
     void engert (int);
     void delete ();
     int queue[10], f=-1, r=-1;
     void mater ()
      int value doice;
       whele (1)
       printf ("InIn ** * MENU* * * In");
       printf ( 1. Inscritioning. Deletioning. Print Reverse in
                4. print Alternatives 5. Exit 1).
       print (In enter your choicen):
       scarf ("tod", sralue);
        street (value):
        ensert (choice):
```

```
call 1: print { ("enter value to be inserted!);
       scanf ( '4.d', & value);
                                  1 11 1
        insert (value);
        break;
case 2: delete();
        break;
 case &; priot-1 ("The reversed queen 18");
         tor(int i = SIZE ; i >0; i --)
    S
                                  PARTIE ALLE
        if (queul[i] == 0)
        continue;
        printf(" · r.d', queue [i]);
                                 1 Williams in
      break;
car H: print-f('Alternative Elements of the queue are: ');
      for (int 1=0; i(SIXE; i+=2)
      1f (queue [1] ==0)
     continue
       printf ("1.d", queue[i]);
      preak;
 can 5 : exit (0);
 refault: printf ("to wrong Selection!!! Try again!!!"):
   y
```

```
333
 Void delete ()
                                      . H Fort
   Pf (f==-1)
    printf ("In Quee is Empty !!! Detection is not possible!!!"):
   else ;
    printf ("In deleted: "I.d", queue[f]);
     f=(f+1) 1. 812E
     8+(f==8)
                          LOF ALL WILLIAM
   f=f=8=-1;
                                   (11) Sti
                 total at the same
                             101-11-11
Difference between Array & Lenked List: The Major defference
between array and Linked List viegards to their structure.
```

Difference between Array & Kenked Kest: The major completely between sarray and kinked hist viegands to this structure between sarray and kinked hist viegands to this structure arrays are index based auta structure where each element and linked hist vietabled with an under. On the other hand, linked hist vietes on references where each mode consists of data and the references to the pervious & next Element.

(5)

```
(5)
    # Proclude 18td 10.h>
(0)
     # Proclude cetal&b.h>
     Struct Mode
    E ent data:
      struct Mode * next;
                   - (Minch , Market 1, Kill
     void print hist (struct Mode * read)
     of
        struct Nocle *ptr=head
        While(ptr)
         print+(" 1.d -> ", ptr-> data);
         ptr= ptr -> next;
    tiple in the state of the ways for
      printf("NULLID");
   4
                      and the same of the
   void push (struct mode ** head, int data)
     struct Mode * new Mode = (struct Mode *) mallor 8138001
                                   struct Mode)):
     new Mode - data = data;
     new Mode -next = * head;
      * head - new Mode;
   4
```

```
void Move Nade ( struct Node #+ dest Ref, struct Node ** DWICE
                                                 Ref)
   If (* sowice Ref == NULL)
      return;
Struct Node * new Mode = * sowice Ref
  * source Ref = (* source Ref) -> next;
  new Alade - next = * dest Ref;
   * deet Ref = new Node;
PM main (void)
  Int keye[]= {1, 2,3}:
  int n = 13000 ( Keys) / 813000 ( Keys [0]);
  structu Node* a= NULL:
 for (int 1= n-1;1 >= 0;1-)
     push (fa, Keys [9]):
struct Node* b = NULL
 for (not 9 = 0; 1<0; 1++)
      pust ($b, 2* Keys[1]);
    Move Made (sa, sb):
   print (" First Listlo"); print (ist (a);
   print (" second Lietln'); printhiet (b);
   return o;
                                        Scanned with CamScanner
```