```
M. Roop Sagar
                                      AP19110010376
1 Write a program to insert and delete an element of the
                      in a linked list where n and k
  nth and kth position
  is taken from user
sol # include (stdio.n)
    # include (stalib.h)
    struct node
                                 Masin
       struct node * next;
                               Le sui silan ;
      3;
      struct node * curv, *temp;
       void input (struct noder)
       void delete (struct node")
       void main (void)
        struct node * S; I have a last many had in
        int n;
        S= Null;
                     MAIN = I draw a really since
        do
       2
        Printf ("Enter the element to insert; \n;");
        print f (" a. Delete \n");
        print f (" 3. Exit \n");
         print of (" Enter the choice:");
                ides ("Fater the number file
                illy and Fibers area
```

```
Scan f (" % d, dn);
  Switch (n)
         Case 1: input (s);
               break;
          case 2: input (s)
                break ;
           } while (n: =3)
                               : table about the
     3
    void input (struct node * x)
    4
                                Think water Liev
     int pos, cz1
     CUTT I Z;
     print f ("Enter the element to be inserted:");
     Scanf (" 1. d", dpos);
                                      ( Nob)
         while (vur -> next! = Null)
         C++; "recalled desired, and reday" begins
         if (c = = pos)
         temp = (struct node *) molloc (size of(struct node))
          printf ("Enter the numbers:");
          scanf (10/0 d', of temp ->n);
```

```
temp -> next = curr -- next;
        curr - next = temp;
        break;
    55 - July - 4 & B - MAI, - d & SPECIATION
              $ $ 1111 - 11110 L
void delete (struct node+z)
  int pos , (= 1)
  curr = 2;
  print f ( Enter the element to be delete!
  scanf ("or d", of pos);
 while (curr -> next!= Null)
  C++;
  if (c = = pos)
  temp = current -> next;
  curr -> next = curr -> next -> next;
  free (temp)
curr = curr -> next;
```

```
void merge (struct node * p, struct node * 9)
  struct node * p-curr=p, * q-curr=*q
  struct node * P_next, *q-next
  while (P_ curr = Null & # 9- curr 1 = Null)
  3
    P_next = P_ curr -> next;
    q-next = q- curr -> next;
    9- curr -> next = P_ next;
    P_ curr -> next = q_ curr;
    p_ curr = p_ next;
    9_ curr = q_next;
   * 9 = 9 - curr
  int main ()
    struct node * P = Null, * 9 = Null;
    push (& P,1);
    push (& p,2);
```

push ( & p; 3); print f ("first linked link: \n") printf (ist (R)); Push ( & 9,4) push ( & 9,5) carcibal abulanta push ( & 9,6) printf ("second linked list: \n"); printf list (9); merge (P,dq); printf ("modified first linked list = \n"); Print List (P); print f ("modified second linked list = \n") void more inche (strace non bier print list (a); Spran barrow & shop tone return 0;

Ellow - Apple - Anom

Action and the second

many also many

struck rade a boils

```
(2) Construct a new linked list by merging alternatives notes
 Of two lists for example in list to the have $1,2,3} and
 ûn lût 2 we have {4,5,6} ûn the new list we should
 have {1,4,2,5,3,6}
Sel # include (stdio.h)
   # include <stalab.h>
    # include Cassertin> 101 (and broad ) is taken
     Struct node
      int data;
      struct node * next;
    yoid move node (struct node ** x; struct node * * y);
     struct node * sorted merge (struct node * 0, struct
     struct node dummy
     struct node * toil = & dummy;
     dummy . next = Null;
     while (1)
      if (a = = Null)
```

```
* y = mewnode -> next;
       new node -> next = *2;
         * 2 - new node;
                            (Hub) ad)
  3
void push (struct node * * head-ref, int new-data)
 struct node * new - node = (struct node *) malloc
                          (sixe of (struct node));
new - node -> data = new-data;
new - node -> next = (* head - ref);
(* head - ref) = new = node;
 void point list (struct node + node)
  While (node 5= Null)
    prints ("%d", node is data);
node = node -> next
   }
 7
```

```
tail -> next = bi
                             break;
                                                                                                    The state of the s
              elseif (b= Null)
tail -> next = a;
                        break;
                                                                                  river i dex new - node - / .....
      if (a -> data <=b -> data)
                 move node (+ (tail) -> next), &a);
                   4
         else
                       move node" (4 (tail) -> next, 46);
                    tail = tail -> next;
                   3
                return (dummy next);
             void move node * (struct node ** 2, struct node * * y)
                      struct node * new node = * y;
                       assert (new node != Null);
```

```
int main ()
       ( 1 MAN CAMP IN MANY AND A MANY DATE OF A LINE
Struct node * nes = null;
 struct node * a = Null;
                        id to a the a leaf of the
 struct node * b = Null;
 push ( 4 a, 1);
  push (4a,2)
  push (& a, 3)
  push (&a,4)
  push (4 a, 5)
  push (& a, 6)
  res = sorted merge (a,b);
  print f ("merge linked list is: \n");
  print list (res);
  return o;
}
                          (L this) of the 1" In
```

Scanned by CamScanner

```
3 Find all the elements in thate stack whose sum is
 equal to k ( where k is given from user)
dol # include < stdio.h >
    int Si[10], top, =-1, S, [10], top 2=-1
                     the grant there
    int si empty 1)
                                  (4,4) 1 11.63
     if (top 1== -1)
        return o,
                                  (+, 1 S) Adul
     int S, top ()
                                  COLL & YILLIY
      return Si [top 1]
                De la ser de l'assert con est l'alle de l'anne
                              2(2ar) 221 24113
       int si pot ( )
      imtop 1 -- ;
      int si push (int x)
        SI[++ top I] = X
        int Sa empty ()
```

```
(1- = = 2 qot) 4i
      return;
   else
    return 0,
     Company of the same
int satop()
return Sa [top 2];
3
 int Sa pop()
d
 3
 int Sa push (int a)
 Sa [++ top 2] = x;
  int Sum [int k)
  int x;
 while (Siempty ()!=1)
   x = sitop ();
   SI pop ();
```

```
While (si empty ()! = 1)
        if (x+sitop ()=k)
         print f ( 1. d, 1. d) \n", a, sitop ();
         Sapush (si top ());
          si pop();
       While (S2 empty ()!=1)
         SI push ( sa top ());
         So pop ();
int main ()
d
 int nie, K;
 print f ("enter the no. of elements of stack: \n");
 scanf ("% d", 4n);
```

```
for (i=0; kn; i++)
        scanf ("10 d", & e);
                                Andrew Milate State of the
         sipush (e);
       printf ("enters the value of constant sum: \n");
       scanf ("% d", 4 K);
       Printf (The combinations whose sum is equal
                        to k is: \n");
        sum(K);
                        sinuet stack as:
                           and the followiting
                         (on malos ) & dians
                         HALL MAIN WELL AUT
                        raint MEnter values
                   Scan & (" L. 1 ") 20002
       That is a sub-
```

Marchano

```
1 white a program to point the elemente in a queue
     i) in neverse order
                             Wall The Man
     ii) inalternate order.
bol: (i) # include < Stdio.h>
  # include "Stack.h"
                          such l'saters ins voins
        # include "QQ.h"
                             int main () and entirely and a saire
          int .n, arr[20], i, j =0;
          struct stack as;
          initstack (.45);
           print f ("Enter no")
            scanf (" % d", fn);
           for (i=0,ikn,i++)
           of
             print ("Enter values:")
              scanf ("1.d", & arr (i);
             for (i=0; i < n; i++)
              insert (arr[i]);
```

3

```
While (i!=n)
          push (ds, dell));
                   It was tour
         printf ("Reverse is");
          While (stop )=-1)
          of
           printf ("1.d", pop(&s));
    print f ("In"); - short source
     return 0;
   # include < stdio.h>
11)
   # include (stdlib.n)
    struct node (
      in) data;
       struct node * next;
   void print nodes (struct node * head)
                They show a show sound
```

Scanned by CamScanner

```
int count = 0
      while (head ! = Null) {
          if (count 7.2 = = 0) {
            printf (" 1. d", head -> data);
             Count ++;
            head = head → next;
void push [struct Node ** head-ref, int new-data]
     struct noder new - node = (struct node *)
   3
                        mollo c (size of (strucf node));
     new-node -> data = new-data;
     new- mode -> next = (* head- ref);
      (* head-ref) = new-node ,
                         是有大部门 安 如何 11 五下1
    int main ()
                  si a oto: Eusti value dans his
      struct node * head = Null;
```

push ( & head , 12); push (& head, 29); push / f head; 11); push / & head, 23); push (& head, 8); printf node (head); return 0; summer and mile al about who autust water him issociated with in witer. to include scidios at Short stands row x a other (430373) with gier

- (ii) how array is different from the linked list

  (ii) white a priogram to add the first element of one list to another list of example we have (1,2,3) in list 1 and (4,5,6) in list 2 we have to get in list 1 and (4,5,6) in list 2 we have to get (4,1,2,3) as output for list 1 and (5,6) for list 2.
- regards to their structure. Averays are indea last based data. Structure where each element associated with an index. On the other hand, linked list relies on reference to the previous and next element
  - tinclude <stdio.n>

    # include <stdio.n>

    # include <stdlib.h>

    struct node

    int data;

    struct node \* next;

void push (struct node \* \* head-ref,

```
int new-data)
struct node * new-node = (struct node x) malloc
                     (size of (struct node));
  new - node - data = new - data;
  new-node -> next = [shead-ref);
     (* head -ref) = new-node;
3
 void printf list (struct node * head)
  Struct node * temp = head;
  white (temp! : Null)
    printf ("%d", temp-)data);
    temp = temp -next;
   print f (" $ \ n");
```