DATA STRUCTURES ASSIGNMENT-4

CHANDANAM TANMAYEE

AP1911,0010402

```
1 Write a program to insert and delete an element at the 11th and
 kth position in a linked list Where n and k is taken from
 user.
 #include <stdio.h>
#include < malloc. h>
#include <stalib.h>
                                           ( ) porgetty
 Struct node ?
   int Value;
    struct node * next;
 y;
  Void insert();
  yoid display ();
   void deletel);
   int count ();
   typedef struct node DATA_NODE;
   DATA_NODE *head_node, * first_node, * temp_node=0, * prev_node
 next_node;
    int data;
    int main() {
      printf ("singly linked list Example - All operations in");
      int option = 0;
       While (option < 5) {
         printf (" in options in");
         printf ("1: Insert into. Linked List In");
         printf ("2: Delete from linked list In"); 1
         printf (" 3: Display linked List In");
          prints ("4: count Linked List In");
```

```
( others: Exit () 10");
                                 CAMITTONOSTE FINE
      print ("Enter your opinion: ");
      Scanf (" >d", boption);
       switch (option) }
       Case 1 monte en ofold bu hour de amper a oficie s
          "insertial); here will be to a mitter and
         case 2:
                                           dd with two shots of the
           delete ();
                                           4 mellom s as marin
            break;
          case 3:
                                         a didlibite made in
            display ();
             break;
          Case 4:
                                               order tol
             count ();
                                     : I can't ohow treet;
             break:
           default:
             break;
       return 0;
   Void insert DE
       printf ("In Enter Element for insert Linked list :h");
        scanf ("".d", & data);
       temp - node = (DATA - NODE *) malloc (Size of (DATA - NODE));
     . temp_node -> Value = data;
        if (first _ node = = 0) ? 1 1011
            first_node = temp_node;
        y else f
                              · ( " of encilous on " \ ) encilon
          helse &
            head - node -> next = temp-node; ...
                       formate and afoling part forms
          temp - node -> next = A; of all a single
            head_node = temp=node
              fflush (stdin);
```

```
2
```

```
Void delete() {
  int Count Value, pos, i = 0;
     Count Value = count () is + this took it include in the
    temp_node = first_node;
                                       Element: In"
     printf ("In Enter Position for Delete
      sconf (" Xd", & pos); - alor . Imple alore and
     if (pos >0 & & pos <= count value) {
       if (pos = = 1) }
         temp_node = temp_node > next;
         first_node = temp_node;
         printf ("In peleted Successfully Inla");
        ય
        else
          while (temp_node (= 0) {
 (income) . if (i== Lpos-1)) {
              prev_node ->next = temp_node ->next;
             if (i = = (countoble -1))
                     head _node = prev_node;
              printf ("In Deleted Successfully In In"); . . .
              break;
             else 5
               i++;
              pier _ node = temp _ node;
              temp_node = temp_node -> next;
                             hadrid transfer of ranger
        3
       else
        printf ("In Invalid Position In 10");
```

```
void display () {
      int count = 0;
      temp_node = first _node;
       printf ("In Display Linked List: 10"); ...
       While (temp_node!=0) ?
         printf ("# xd# " temp_node > value);
          count ++;
          temp_node=temp_node >next;
       printf ("In No of items in linked list: Xd In (ount);
    3
     int count () {
         int count = 0;
         temp_node = first_node;
         While (temp-node != 0) {
             count ++:
            temp_node = temp_node ->next;
        prints ("In No of Hems In Linked List: Yd In", count);
         return (ount;
Output:
1: Insert into Linked List
2: Delete from linked list
3: Display linked list
4: Count Linked list
others: Exit 1) . ....
Enter your option: 1
Enter Element for Insert Linked list:
3
options
1: Insert into Linked list
2: Delete from Linked list.
```

```
3: Display Linked List
 4: Count Linked List
                      (often help ben 1 * 4 chest
 others: Exit()
Emer your optionis. ( Totals land) - should be on
Display Linked list:
No of Items in Liked List: I had the house stokeness
options is some
1: Insert into Linked List r Kahan mutal appropriation
2: Delete from Linked List
3: Display Linked List
4. Count Linked List
others : Exit ()
Enter your option: 6.
2) Construct a new linked list by merging alternate nodes of
two lists for example in list I we have {1,2,33 and in
list 2 we have {4,5,69 in the new list we should have
 81,4,2,5,3,69
 #include <stdiah>
 # include < stdlib.h>
Struct Mode
٤
       int data;
       struct Node* next;
 void printlist (struct Node* head).
 y;
 ٤
       smuct Node* pt = head;
        while(pt1)
              printf (" y.d -> ", ptr -> data);
        ક્
                b# = b# > vext.). ( ) innop such.
          y
```

```
printf ("NULLIN");
Void push (struct Node * * head , int data)
    Struct Node* newNode = (struct Node*) malloc (size of (struct Node));
٤
        newNode >> data = data;
         newNode -> next = *head; 1: bis head a suit for
          * head = new Node;
      Node* shuffle Merge (struct Node* a, struct Node* b)
z
٤
      Struct Node dummy;
       struct Node* tail = &dummy;
dummy next = NULL;
  ... while (1).
   a ros Ephose ; for a = = NULL)
                 +ail -> next = b;
                  break;
              else if (b=NULL)
                 tail -> next=a;
                  break;
               ч
              els e
                  tail -> next =a;
                    tail=ashbaitalest but is tailing
                   +ail -> next = b;
                     tall = 5;
                     b=b -> next;
            3
              return dummy next;
```

```
int main (void)
 કૃ
                                                 I Tien for
      int keys [] = {1,2,3,4,5,6,7 };
     int n = site of (keys) /site of (keys [07);
       Struct Node *a=NULL; *b=NULL;
        -for (int i = n-1; i7=0; i=i-2)
              push ( ka, keys [i]);
         for (inti= n-2; i >=0; i= i-2)
                push ( & b , keys [i]);
           printf ("first list; ");
            print List (a);
           printf ("Second List; "); '''
            print List (b);
            struct Node* head = Shuffle Merge (0,6);
             printf ("After Meige: ");
              print List (head);
             return D; (+ + 1: hours );
 OUT PUT !
First list: 1 -> 3 -> 5 -> 7 -> NULL
Second List: 2 >4 >6 -> NULL
 After Merge: 1->2->3->4->5->6->7->NULL
3) Find all the elements in the stack whose sum is equal
to k (where k is given from wer),
#include <stdio. b>
 int top = -1;
 int X;
 char stack [100];
                 " " or! " ! llog of dants ... " I thing
 void push (int x);
```

```
char pop ();
                                             (Bery siber to
int main 1)
                        1 16 A 3 H - 1 1 4
  int i, n, a, t, k, f, Sum = p, (ount = 1;
  printf ("Enter the number of elements in the stack"),
   scanf ("Yd", kn);
                          1-110 -111 -1 +411 -1
   for (i=0; i<n; i++){
    printf ("Enter next element"),
    Scanf (" ", d", ka);
     push (a);
    printf ("Enter the number of to be checked");
     scanf ("Y.d", & K); ....
      for (1=0; i<n; i++)
         t = bob ();
          Count + = 1;
          if (sum = = K) {
           for (int j=0; j < (ount ; j++) id 11 11 12.
           printf ("Yd", stack[j]);
           9=1;
           break;
         push (4);
        if (f[=1)
       printf ("The elements in the stack clost add up to the Sum");
     void push (int X)
       if (top==99)
         printf ("In Stack is full!!! In"); ilent
         return;
```

```
top = top, to, in so the section of the contract of the section of the section of
    Stack [top] = Xi.
                  The first twee to be the control of the control
    char pop ()
       if (stack [top] == -1)
     print f ("In Stack is EMPTY!!! (n"));
                                (nuley a ' h ; ' ) furni
         return o;
       X = Stock [top];
       top=top-1;
           return X;
    ٠
٠
OUT PUT:
Enter the number of elements in the stack 4
 Enter next element 3
  Enter next element 1
   Enter next element 8
   Enter next element9
   Enter the sum to be checked 21
   The elements in the stack don't add up the sum
                   Windson Steering AN "1 4
4) Write a program to print the elements in a queue
   i. in reverse order
  ii.in alternate order.
 #include <stdio. h> noor ', ....
 # define SIZE 10
 void insert (int);
  Yord deletert);
  int queue [10], f = -1, x = -1;
   void main () {
      int value , Choice;
```

```
While (1) 9
   printf ("1. Insertion In 2. Deletion In 3. Print Reverse In 4. Print
                            - Alterate In 5 Exiting I hat
    printf ("In Enter your choice: ");
     Scanf (" Xd", & choice);
                                  /1- - - ( got of donte) gi
      switch (choice) &
      case 1: printif ("Enter the value to be insert;");
        Scanf (" y.d", &value);
        insert (value);
        break;
       Case 2 : delete U;
        break;
       Case 3:
                printf ("The Reversed queue is: ");
                 for (int i = SIZE ; i == 0; i --)
                                Almondo in rooms of some
        ફ
               if (queuc[i] ==0)
                   continue;
                 printf (" >d", queue [i]);
             break in hinh
           printf ("Alternate elements of queue are:");
                for (in+ 1=0 ; i < Si = En; i += 2) org
        3
               if (queue [i] = =0)
                   continue;
                  print & ("y.d", queue (i)); with the observation
               break;
         default: printf ("In Wrong selection!!! Thy again!
        Case 5: exit (0);
           y
                                      , 9 : 10113 Only 1011
      33
```

```
Void insert (int value) {
    if(f==0 & & r=512E-1)11 f==++1)
       printf ("In Queue is full!! Insertion is not possible!!");
      else §
          if (f = = -1)
        f=0;
          1=(1+1)%512=;
           queue [r] = value;
            printf ("In Insertion Success!!");
    33
  Void delete () {
     if (f == -1)
       printf ("In Queue is Empty!! Deletion is not posible!! ");
      else ?
       printf ("In Deleted: Y.d", queue [f]);
         f = (f+1) % SIZE;
         if (f = = 1)
        f=r=-13.
          industry the second
OUTPUT :
 1. Insertion . but war
 2. Deletion
 3. Print Reverse
 4. Print Alternate
5. Exit
Enter your choice: !
Enter the value to be insert: 5
 Invertion Success!!!
1. Insertion
 2. Deletton
 3. print Kevere
```

& (sulur fat) tosait his 4. Print Alternate (HI - 7110-1812-1420 7) CH 5. Exit, Enter your choice while authorist that I survey out I to see Enter the value to be insert; 3 Insertion Success!!! 1. Insertion 3 8/30/0 (14 6) . 2. Deletion Dular : (1) enoug 3. Print Reverse 4. Print Alterate 5.EXH Enter your choice:3 The Reversed queue is: 3 5 ersea y ... 1. Insertion 2. Deletion 3. Print Reverse : ([] annug iby : brilding 4. Print Alternate 1 1 412 30 1173 3 5. Exit Enter your choice: 5 (1 -719: Program - 5; (i) How array is different from the linked list. Sol: The major difference between array and linked list regards to their structure. Arrays are index based data structure, Arrays Where each element associated with an index while a linked list is a dola structure which contains a sequence of the elements where each element is linked to its next element. (1i) Write a program to add the first element of one list to another list for example we have {1,2,3 } in list 1 and {4,5,6} in list 2 we have to get {4,1,2,3} as output for list I and 25,64 for list 2.

```
# include <stdio.h
# include <stdlib.h>
Struct Node
§
    int data;
     struct Node * next;
 3;
roid print List (shurt Node * head)
ş
    Struct Node* ph = head;
     while (pti)
          printf ("xd" -> ", pt+ -> data);
           pt = pt -> next;
       prints ("NULL In");
void push (shuct Node ** head, int data)
    Struct Node* new Node = (struct Node*) malloc (size of (struct Node));
      newNode →data =data;
       newNode -> next = *head;
          * head = new Node;
    MoveNode (struct Node** destref, struct Node* * source Ref)
void
  ٤
         if (* source Ref == NULL)
          Return;
          struct Node* new Node = * source Ref;
            * SourceRef = (* SourceRef) -> next;
             new Node -> next = * dest Ref;
             * dest ref = new Node;
   3
int main (void)
3
         int keys[] = {1,2,3};
          int n = Size of (keys) / Size of (keys [o]);
```

```
Struct Node *a = NULL;
      for (int i = n-1; 17=0; i--)
            push (ka, key[i]);
         Struct Node* b= NULL
       for (int i = 0; i < n; i++)
            push ( wb, 2* keys [i]);
             MoveNode ( Wa, &b);
            printf (" first list: ");
           printlist (a);
           printf ("Second List: ");
            printList (b);
           return o;
OUTPUT:
First List: 6 →1 →2 →3 → NULL
second list: 4->2-> NULL
```