1) Singly Linked List, sort, reverse, concatination 4 Smut node of int data; Struct node *next. 3 Strut node * head L = NULL; Strut node + head = NULL; Struct node *insert 1 (struct node *head, int num) (Struct node *new_node , *ptr; new-node = (struct node) mullor (street (struct node)); new-node -> date = num. new-node -> next = Null; of if (head 1 = = NULL) (head 1 = new-node; I also of ptr = head 1 . while (ptr->next 1= NULL)? ptr = ptr->next; ptr = next = new-node; return Reheard L

Struct node * tracert2 (struct node *head? int num) Street node *new-node, *ptr; new-node = (struct node) malloc (size of (struct node)) new_node -> data = num, new-node -> next = NULL; if (headle = NULL) (head 1 = new-node ; I else 1 ptr = Lead 2. while (ptr-snext to NULL) of gtr = ptr-next ptr - next = new-node; rehurn head 2; Store concert (struct rode *head 1, struct rode *heade) struct node *ptr; of (head 1 = NULL 91 head 2!= NULL) x ptr = head . while (ptronext != NULL) ptr = ptronest, ptr-next = head 2; I else f prints ("either of the Listed List is empty");

display (struct node * head) & strut node *ptr; if (head = = NULL) { prints (" stackinked List is empty In"); I else { ptr = head; while (ptr != Null) printf ("Y.d", ptr-data). ptr = ptr = next.); void sort (struct rode thead) of struct node * ptr 1, * ptr2; \$ptrl = heard while (ptr + next != xul) ptr2 = ptrbnert; while (ptr2 1= NULL) & if (pril -) data > pri2 -> data) (emp: ptr [-> data . ptri - data : ptr2 - data; pt12 - doita = temp; ptr2 = ptr2 - next; ptrs - ptil - next. diplay (head).

```
void reverse (struct node thead) ?
        At net ande Aprix, Aprel;
          Me kead;
Void
        reperse (struct node + head) f
          Struct node *ptr = NULL , *prer = NULL)
          while (head := MULL) {
               ptr = head > next;
              head > next = prev;
              prev = head.
              heard = ptr;
          head = prev;
        display (head)
   3
  0/p :
    Enter your Choice :1
    Enter the number: 10
    Genter your choice: 1
    Easter the number: 35
    Enter your Choice: 1
    Enter the number: 24
    Enter your choice: 3
     €n 10 → 35 → 24 →
```

Enter your Choice: 2 Goter the number: 32 Enter your choice: 2 Conter the number: 21 Enter your choice: 2 Enter the number: 2 Enter your choice: 4 32->21->2 -> Enter your choice: 5 10 -> 35 -> 24-> 32-> 21-> 2-> Onter your choice: 6 2-10-21-24-32-35 Enter your choice: 7 35-32-24-21-710-72-Ja. Stack tonplementation using single Linked List Strutt node (int data; Struct node - hext; struct node *heerd = NULL. void push L) & Street node * new node. int num;

```
prints ( Enter the number );
scanfluoted "mun);
 hew node = (struct node +) malloc (size of (struct node)
  new-node > data = num;
   if (heard == MULL) {
     head = new-node;
     new, node -> next & NULL;
   Jelse (
      new-node >next = head.
       head = new-node;
Veid populy
    Smut node *ptr;
     If (head == NULL) &
            printf ("Under flow ... In");
        3 else 1
           proshead.
          head = ptr-shext;
           printf ("Poped element: "I'd In", ptrodute);
           free (pti);
```

void display 194 struct node *ptr; ptr = head. printf ("Hements of Linked List: "); while (ptr != MULL) ? printf ("1" -> ", plr >deta); ptr = ptr -> data, printf("/n"); 0/p: Enter your Choice: 1 Enter the number: 10 Enter your choice: 1 Enter the number: 20 Enter your choice : 1 Entery the number: 30 Enter your choice: Element of Linked List : 30 -> 20 -> 10 -> Enter the number: 2 poped element: 30 Goter the number: 2 poped element: 20 Enter the number: 2. poped element: 10

Enter the number: 2 Underflower 26) Queue Implementation using single Linked List Struct node of int data . Struct node * next; 8 Struct node + head = NULL; void engueure () { struct nocle *ptr, *new-node; Int num; prints (" 6ster the number:"); Scanf ("1.d", Shum); new-node = (strut node*) mallor (pire of (strut rod) new-node > data = num; new-node->next = NULL; if I head = = NULI) head = new-node; Jelse L ptr = head; while (ptr -> next 1 c MULL) d ptr = ptronext; 9 ptr mext = new-node

void dequeur() of struct node *ptr; if (head == NULL) & printf ("Under flow .. "): else 2 pler = head; head = ptr-head; printf(" Poped element " Ved In", ptrodata); free (pti); void display () { Smut node *ptr; If (head == plub) & printl("Jueue is Empty ... In"); delse of pto a head; printf ("Hemen's of Linked List: In"); while (ptr := NULL) ? printf ("7.d", ptr >data); ptr = ptr = next. print+ ("In");

0/p: Gntur your choice: 1 frumber: 10 Choice: 1 number: 20 Choice : 1 number: 30 Choice: 3 tlements of Linked List: 10 → 20 → 30 → Choice: 2 dequeux element: 10 Choice: 2 dequire element: 20 Choice: 2 dequeue element: 30 Choice: 2 queue is Empty ...