05-02-24

Q) write to program to implement doubly link list with primitive operations
a) create a doubly linked list.
b) Insert a new node to left of nod
c) Delete the node based on a specifi
value.

→

```
struct node
{
    struct node *next;
    int data;
    struct node *prev;
};

struct node *create_ll (struct node *sta
{
  struct node *new_node, *ptr;
  int num;
  printf ("\n Enter -1 to end :");
  printf ("\n Enter data:");
  scanf ("%d", &num);
  while (num != -1)
  {
    if (start = = NULL)
    {
      new_node = (struct node *) malloc
                    (sizeof (struct node));
      new_node->prev = NULL;
      new_node->data = num;
new_node -> next = NULL;
 start = new_node;
 }
else
 {
   ptr = start;
   new_node = (struct node *) malloc (sizeof(struct
                                              node))

    new_node -> data = num;
    new_~~node~~ node -> next = NULL;
     start =
     while (ptr -> next != NULL)
            ptr = ptr -> next;

      ptr -> next = new_node;
      new_node -> prev = ptr;
       new_node -> next = NULL;
     }
    printf(" \n Enter the data:");
     scanf (" %d ", &num);
   }
   return start;
 }.

struct node *display (struct node * start)
{
    struct node *ptr;
     ptr = start;
```

```
while(ptr != Null)
{
    printf("\t %d", ptr->data);
    ptr= ptr->next;
}
return start;
}

struct node *insert-before (struct node *st
{
    struct node *new_node, *ptr;
    int num, val;
    Printf("\n Enter the data: ");
    Scanf ("%d", &num);
    printf("\n Enter the value before which the data
                    has to be inserted ");
    Scanf ("%d", &val);
    new_node = (struct node *)malloc(sizeof(stru
                                        node));
    new_node->data= num;
    ptr = start;
    while (ptr->data != val)
    {
        ptr = ptr->next;
    }
    new_node->next = ptr;
    new_node->prev = ptr->prev;
    ptr->prev->next = new_node;
    ptr->prev = new_node;
    return start;
}

struct node *delete_position( struct node
                                     * start)
 {
    struct node * ptr;
     int  val ;
     ptr = start;
     printf("\n Enter the value to be
                       deleted : \n");
     scanf (" %d", & val);
     while (ptr->data != val)
      {
        ptr=ptr->next;
      }

     ptr->prev->next = ptr->next;
     ptr->next->prev = ptr->prev;
     free (ptr);
    ~~return start~~ ;
}
```

Output

```
1 Create a list      2. Display the list.
 3. Add a node before a given node
 4. Delete a given node      5. Exit.

 Enter your choice: 1.
  Enter -1 to end
  Enter the data: 2
  Enter the data: 3
  Enter the data: 5
  Enter the data: -1

 Doubly Linked list created

 1. Create a list      2. Display the list.
 3. Add a node before a given node
 4. Delete a given node      5. Exit.
  Enter your ~~to~~ ~~end~~ choice: 2

   2   3   5

  1. Create a list      2. Display the list.
  3. Add a node before a given node
  4. Delete a given ~~node~~      5. Exit
   Enter your ~~choice~~: 3
    Enter the data to be inserted: 4
    Enter the value before which data to be
    inserted: 4
```

```
1. Create a list          2. Display the list.
3. Add a node before a given node
4. Delete a given node          5. Exit.

Enter your choice: 4.

Enter the value to be deleted: 3.

1. Create a list       2. Display the list.
3. Add a node before a given node.
4. Delete a given node       5. Exit.
Enter your choice: 2.


2  4  5.
```

[illegible]
4/3/24