C++ Program to Implement Doubly Linked List

Doubly linked list is a type of data structure that is made up of nodes that are created using self-referential structures. Each of these nodes contain three parts, namely the data and the reference to the next list node and the reference to the previous list node.

Only the reference to the first list node is required to access the whole linked list. This is known as the head. The last node in the list points to nothing so it stores NULL in that part. Also the doubly linked list can be traversed in both directions as each node points to its previous and next node.

A program to implement doubly linked list is given as follows.

Example

```
#include <iostream>
using namespace std;
struct Node {
   int data;
   struct Node *prev;
   struct Node *next;
};
struct Node* head = NULL;
void insert(int newdata) {
   struct Node* newnode = (struct Node*) malloc(sizeof(struct Node));
   newnode->data = newdata;
   newnode->prev = NULL;
   newnode->next = head;
   if(head != NULL)
   head->prev = newnode ;
   head = newnode;
}
void display() {
   struct Node* ptr;
   ptr = head;
   while(ptr != NULL) {
      cout<< ptr->data <<" ";
      ptr = ptr->next;
   }
}

int main() {
   insert(3);
   insert(1);
   insert(7);
   insert(2);
   insert(9);
   cout<<"The doubly linked list is: ";
   display();
   return 0;
}
```

output

```
The doubly linked list is: 9 2 7 1 3
```

In the above program, the structure Node forms the doubly linked list node. It contains the data and a pointer to the next and previous linked list node. This is given as follows.

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

The function **insert()** inserts the data into the beginning of the doubly linked list. It creates a newnode and inserts the number in the data field of the newnode. Then the prev pointer in newnode points to NULL as it is entered at the beginning and the next pointer points to the head. If the head is not NULL then prev pointer of head points to newnode. Finally the head is the newnode i.e. the linked list starts from there. This is given below.

```
void insert(int newdata) {
   struct Node* newnode = (struct Node*) malloc(sizeof(struct Node));
   newnode->data = newdata;
   newnode->prev = NULL;
   newnode->next = head;
   if(head != NULL)
   head->prev = newnode ;
   head = newnode;
}
```

The function **display()** displays the whole doubly linked list. First ptr points to head. Then it is continuously forwarded to the next node until all the data values of the nodes are printed. This is given below.

```
void display() {
   struct Node* ptr;
   ptr = head;
   while(ptr != NULL) {
      cout<< ptr->data <<" ";
      ptr = ptr->next;
   }
}
```

In the function main(), first various values are inserted into the doubly linked list by calling insert(). Then the doubly linked list is displayed. This is given below.

```
int main() {
   insert(3);
   insert(1);
   insert(7);
   insert(2);
   insert(9);
   cout<<"The doubly linked list is: ";
   display();
   return 0;
}
```