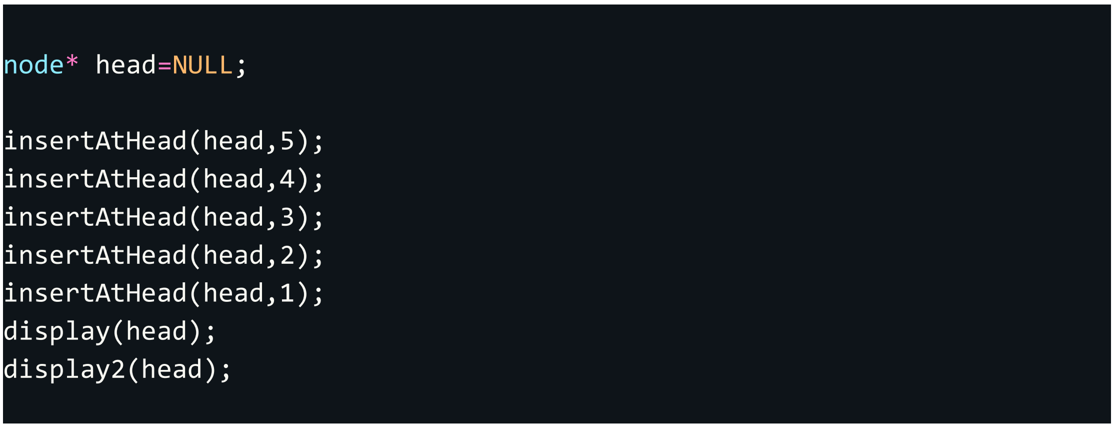
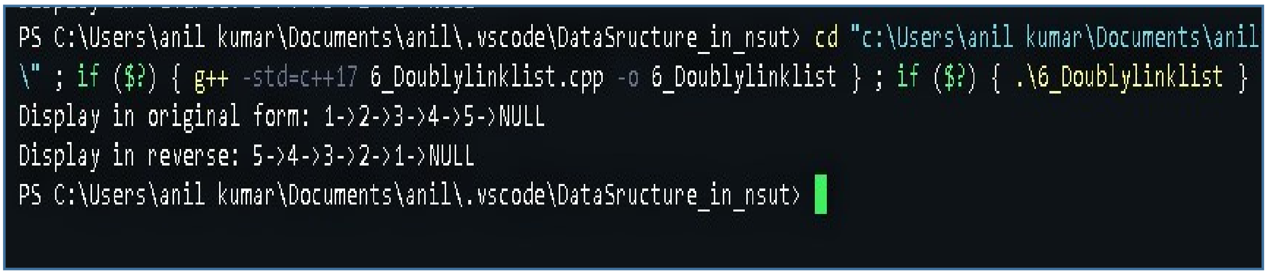
Source Code

```
#include<iostream>
using namespace std;
class node{
public:
    int data;
    node* prev,*next;
    node(int val){
        data=val;
        prev=NULL;
        next=NULL;
    }
};
void insertAtHead(node* &head,int val){
    node* n=new node(val);
    if(head==NULL){
        head=n;
        return;
    }
    head->prev=n;
    n->next=head;
    head=n;
}

void display(node* head){
    if(head==NULL){
        return;
    }
    while(head!=NULL){
        cout<<head->data<<"->";
        head=head->next;
    }
    cout<<"NULL"<<endl;
}
void display2(node* head){
    cout<<"Display in reverse: ";
    if(head==NULL){
        return;
    }
    while(head->next!=NULL){
        head=head->next;
    }
    while(head!=NULL){
        cout<<head->data<<"->";
        head=head->prev;
    }
    cout<<"NULL"<<endl;
}
int main(){
```



```
node* head=NULL;

insertAtHead(head,5);
insertAtHead(head,4);
insertAtHead(head,3);
insertAtHead(head,2);
insertAtHead(head,1);
display(head);
display2(head);
```

Output



```
PS C:\Users\anil kumar\Documents\anil\.vscode\DataSructure_in_nsut> cd "c:\Users\anil kumar\Documents\anil
\" ; if ($?) { g++ -std=c++17 6_Doublylinklist.cpp -o 6_Doublylinklist } ; if ($?) { .\6_Doublylinklist }
Display in original form: 1->2->3->4->5->NULL
Display in reverse: 5->4->3->2->1->NULL
PS C:\Users\anil kumar\Documents\anil\.vscode\DataSructure_in_nsut> 
```