# Lab Task 5

## Code Explanation:

Singly Linked List implementation with functions to display nodes (first, last, at position, center, all).

## Code:

#include <iostream>  
using namespace std;  
  
class Node {  
public:  
 int data;  
 Node\* next;  
 Node(int val) : data(val), next(nullptr) {}  
};  
  
class SinglyLinkedList {  
private:  
 Node\* head;  
 int size;  
  
public:  
 SinglyLinkedList() : head(nullptr), size(0) {}  
  
 void insert(int val) {  
 Node\* newNode = new Node(val);  
 if (!head) head = newNode;  
 else {  
 Node\* temp = head;  
 while (temp->next) temp = temp->next;  
 temp->next = newNode;  
 }  
 size++;  
 }  
  
 void first() { if (head) cout << head->data << endl; }  
  
 void last() {  
 if (!head) return;  
 Node\* temp = head;  
 while (temp->next) temp = temp->next;  
 cout << temp->data << endl;  
 }  
  
 void at(int pos) {  
 if (pos < 1 || pos > size) return;  
 Node\* temp = head;  
 for (int i = 1; i < pos; i++) temp = temp->next;  
 cout << temp->data << endl;  
 }  
  
 void center() { at((size / 2) + (size % 2)); }  
  
 void all() {  
 Node\* temp = head;  
 while (temp) {  
 cout << temp->data << " ";  
 temp = temp->next;  
 }  
 cout << endl;  
 }  
};  
  
int main() {  
 SinglyLinkedList sll;  
 sll.insert(10);  
 sll.insert(20);  
 sll.insert(30);  
 sll.insert(40);  
 sll.insert(50);  
  
 sll.all();  
 sll.first();  
 sll.last();  
 sll.at(3);  
 sll.center();  
  
 return 0;  
}

## Output:

```  
10 20 30 40 50  
10  
50  
30  
30  
  
```