#include <iostream>

using namespace std;

struct node {

int x;

node \*next;

};

int main()

{

node \*f; // This won't change, or we would lose the list in memory

node \*y; // This will point to each node as it traverses the list

f = new node; // Sets it to actually point to something

f->next = NULL; // Otherwise it would not work well

f->x = 12;

y = f; // The conductor points to the first node

if ( y != NULL ) {

while ( y->next != NULL)

y = y->next;

}

y->next = new node; // Creates a node at the end of the list

y = y->next; // Points to that node

y->next = NULL; // Prevents it from going any further

y->x = 42;

y = f;

if ( y != NULL ) { //Makes sure there is a place to start

while ( y->next != NULL ) {

cout<< y->x<<"\n";

y = y->next;

}

cout<< y->x;

}

// Insert at first

cout<<"\n ------------- \n";

y = new node;

y->x = 55;

y->next = f;

f = y;

y = f;

cout<<"\n";

while ( y != NULL ) {

cout<< y->x<<"\n";

y = y->next;

}

cout<<"\n ------ INSERT AT LAST ------- \n";

node \*z;

z = f;

while ( z->next != NULL ) {

z = z->next;

}

// cout<<"Z = "<<z->x<<"\n";

y = new node;

y->x = 155;

y->next = NULL;

z->next = y;

// cout<<"Y = "<<y->x<<"\n";

y = f;

cout<<"\n";

while ( y != NULL ) {

cout<< y->x<<"\n";

y = y->next;

}

// Delete from first

y = f;

f = y->next;

delete(y) ;

y=f;

cout<<"\n";

while ( y != NULL ) {

cout<< y->x<<"\n";

y = y->next;

}

// Deletion from last

node \*s\_l = f;

while (s\_l->next->next != NULL)

{

s\_l = s\_l->next;

}

delete (s\_l->next);

s\_l->next = NULL;

y = f;

cout<<"\n";

while ( y != NULL ) {

cout<< y->x<<"\n";

y = y->next;

}

}