**Dynamic linked list**

// linklist.cpp

// linked list

#include <iostream>

using namespace std;

////////////////////////////////////////////////////////////////

struct link //one element of list

{

int data; //data item

link\* next; //pointer to next link

};

////////////////////////////////////////////////////////////////

class linklist //a list of links

{

private:

link\* first; //pointer to first link

public:

linklist() //no-argument constructor

{ first = NULL; } //no first link

void additem(int d); //add data item (one link)

void display(); //display all links

};

//--------------------------------------------------------------

void linklist::additem(int d) //add data item

{

link\* newlink = new link; //make a new link

newlink->data = d; //give it data

newlink->next = first; //it points to next link

first = newlink; //now first points to this

}

//--------------------------------------------------------------

void linklist::display() //display all links

{

link\* current = first; //set ptr to first link

while( current != NULL ) //quit on last link

{

cout << current->data << endl; //print data

current = current->next; //move to next link

}

}

////////////////////////////////////////////////////////////////

int main()

{

linklist li; //make linked list

li.additem(25); //add four items to list

li.additem(36);

li.additem(49);

li.additem(64);

li.display(); //display entire list

return 0;

}