#include <iostream>

#include <conio.h>

using namespace std;

struct Node // node definition

{

int data;

Node \*next;

};

struct Node \*head,\*p,\*tail; // globle variable

void InsertNodeAtTheFirstPostion(int value) //function

{

Node \*temp = new Node;

temp->data = value; // insert at the first position

temp->next = head;

head = temp;

}

void InsertAtTheEndPosition(int value)

{

Node \*temp = new Node;

temp->data = value;

if(head == NULL) // when linked list is empty

{

temp->next = head;

head = temp;

}

else // when linked list have nodes

{

p = head; // p is temporary pointer variable

while(p->next != NULL)

{

p = p->next;

}

p->next = temp;

temp->next = NULL;

}

}

void InsertAtTheNthPosition(int pos,int value)

{

Node \*temp = new Node;

temp->data = value;

temp->next = NULL;

if(pos == 1) // insert at the first position

{

temp->next = head;

head = temp;

return;

}

Node \*temp2 = head; // temp2 is temporary pointer variable

for(int i = 1; i < pos - 1; i++)

{

temp2 = temp2->next;

}

temp->next = temp2->next;

temp2->next = temp;

}

void PrintNode()

{

Node \*temp1 = head;

if(temp1 == NULL)

{

cout << "list is empty";

}

else

{

cout << " List : ";

while(temp1 != NULL)

{

cout << " " << temp1->data;

temp1 = temp1->next;

}

cout << endl;

}

}

void main()

{

head = NULL;

tail = NULL;

int ch,value,pos;

while(1)

{

cout << "1 for insert node at first \n2 for insert node at end \n3 for insert node at the nth node \n4 for display the node \n5 for end the program \n";

cin >> ch;

switch(ch)

{

case 1:

cout << "Enter the node data ";

cin >> value;

InsertNodeAtTheFirstPostion(value);

break;

case 2:

cout << "Enter the node data ";

cin >> value;

InsertAtTheEndPosition(value);

break;

case 3:

cout << "Enter the node data ";

cin >> value;

cout << "Enter the position of the node ";

cin >> pos;

InsertAtTheNthPosition(pos,value);

break;

case 4:

PrintNode();

break;

case 5:

exit(0);

break;

case 6:

cout << "Wrong Choice";

break;

}

}

getch();

}