Data/Item node \* next/link

#include <iostream>

using namespace std;

struct node {

int data;

node \* next; //link or right

};

void main() {

node \* P, \*Q, \*T, \*Z;

P = new node;

P->data = 10;

P->next = NULL; //0

Q = new node;

Q->data = 20;

Q->next = NULL; //0

T= new node;

T->data = 30;

T->next = NULL; //0

Z = new node;

Z->data = 40;

Z->next = NULL; //0

P->next = Q; // (1)

Q->next = T; // (2) P->next->next = T;

T->next = Z; // P->next->next->next = Z;

//cout << P->next->next->next->data << endl; // Z->data

P->next->next->next->next = new node;

P->next->next->next->next->data = 50;

P->next->next->next->next->next = NULL;

while (P != NULL) {

cout << P->data << endl;

P = P->next;

}

system("pause");

}

#include <iostream>

using namespace std;

struct node {

int data;

node \* next; //link or right

};

void main() {

node \*Head, \* P;

Head = NULL;

for (int i = 0; i < 5; i++) {

P = new node;

P->data = rand() % 100;

P->next = NULL;

if (Head != NULL) {

P->next = Head;

Head = P;

}

else Head = P;

}

P = Head;

while (P != NULL) {

cout << P->data << endl;

P = P->next;

}

cout << " deletionn \n";

// Delete Front node

P = Head;

Head = Head->next;

delete P;

P = Head;

while (P != NULL) {

cout << P->data << endl;

P = P->next;

}

system("pause");

}

#include <iostream>

using namespace std;

struct node {

int data;

node \* next; //link or right

};

node \* Head;

void Insertfront() {

node \* P;

P = new node;

P->data = rand() % 100;

P->next = NULL;

if (Head != NULL) {

P->next = Head;

Head = P;

}

else Head = P;

}

void Deletefront() {

node \* P;

if (Head != NULL) {

P = Head;

Head = Head->next;

delete P;

}

}

void Printlist() {

node \* P;

P = Head;

while (P != NULL) {

cout << P->data << endl;

P = P->next;

}

}

void main() {

Head = NULL;

for (int i = 0; i < 10; i++) {

Insertfront();

}

Printlist();

Deletefront();

cout << endl;

Printlist();

system("pause");

}