**ЗВІТ**

**до лабораторної роботи № < 12.4 >**

**« Опрацювання кільцевого однонаправленого списку »**

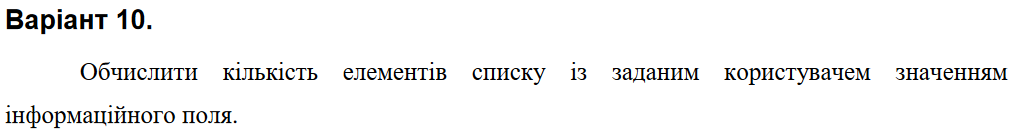
**з дисципліни**

**«Алгоритмізація та програмування»**

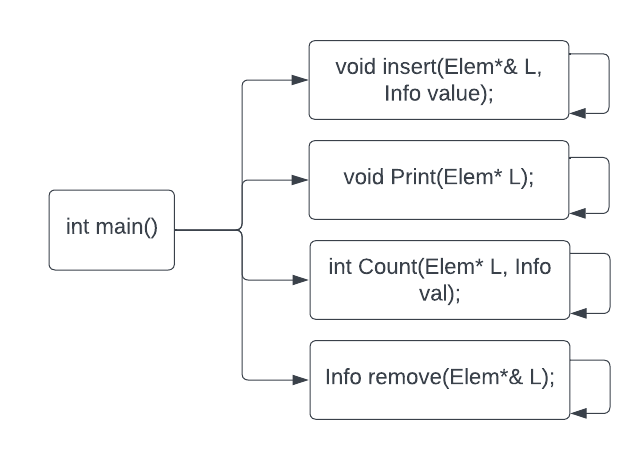
**студента групи ІН-105Б**

**Горанова Анастаса**

**Умова завдання:**



**Структурна схема:**



**Текст програми:**

// 12.4REC.cpp

// < Горанов Анастас >

// Лабораторна робота № 12.4REC

// Опрацювання кільцевого однонаправленого списку

// Варіант 10

#include <iostream>

#include <time.h>

using namespace std;

typedef int Info;

struct Elem

{

Elem\* link;

Info info;

};

void insert(Elem\*& L, Info value);

void remove(Elem\*& L);

Elem\* lastElem(Elem\*& L, Elem\*& T);

int Count(Elem\*& T, Elem\*& first, Info k, Info val);

void Print(Elem\*& L, Elem\*& first);

int main()

{

srand((unsigned)time(NULL));

Elem\* L = NULL;

insert(L, 0);

Elem\* h = L;

Elem\* first = L;

Print(h, first);

Info val;

cout << endl;

cout << "Enter the value of the element: "; cin >> val;

cout << endl;

cout << "Found " << Count(L, first, 0, val) << " elements with value " << val << endl;

remove(L);

cout << endl;

}

void insert(Elem\*& L, Info n)

{

Info value = -10 + rand() % 20;

Elem\* tmp = new Elem;

tmp->info = value;

if (L != NULL)

{

Elem\* T = L;

T = lastElem(L, T);

T->link = tmp;

}

else // Якщо список порожній

{

L = tmp;

}

tmp->link = L;

if (n < 10) {

n++;

insert(L, n);

}

}

Elem\* lastElem(Elem\*& L, Elem\*& T) {

if (T->link != L) {

T = T->link;

return lastElem(L, T);

}

else {

return T;

}

}

void remove(Elem\*& L)

{

Elem\* T = L;

T = lastElem(L, T);

Info value = L->info;

if (T != L)

{

Elem\* tmp = L->link;

delete L;

L = tmp;

T->link = L;

}

else

{

delete L;

L = NULL;

}

cout << value << " ";

if (L != NULL) {

remove(L);

}

}

void Print(Elem\*& h, Elem\*& first) {

if (h == NULL)

return;

if (h->link != first)

{

cout << h->info << " ";

h = h->link;

Print(h, first);

}

else {

cout << h->info << " ";

return;

}

}

int Count(Elem\*& T, Elem\*& first, Info k, Info val)

{

if (T == NULL)

return 0;

if (T->link != first)

{

if (T->info == val)

k++;

T = T->link;

Count(T, first, k, val);

}

else

{

if (T->info == val)

k++;

return k;

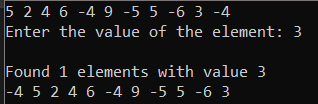
}

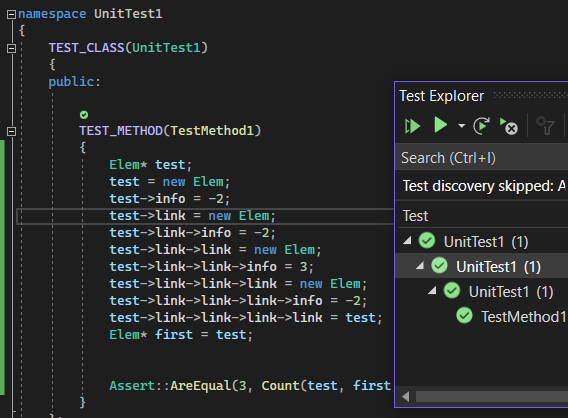
}

**Посилання на git-репозиторій з проектом:**

*https://github.com/StassNG/12.4REC*

**Результати програми та unit-тесту:**

****

****

**Висновок:** я навчився опрацьовувати кільцевий однонаправлений список.