**ЗВІТ**

**до лабораторної роботи № < 12.2 >**

**« Опрацювання лінійного однонаправленого списку »**

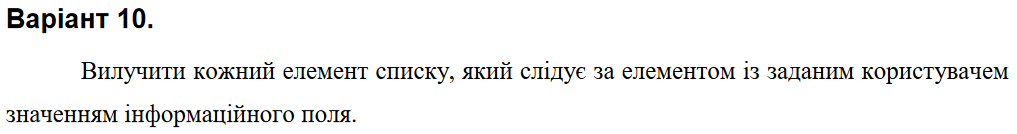
**з дисципліни**

**«Алгоритмізація та програмування»**

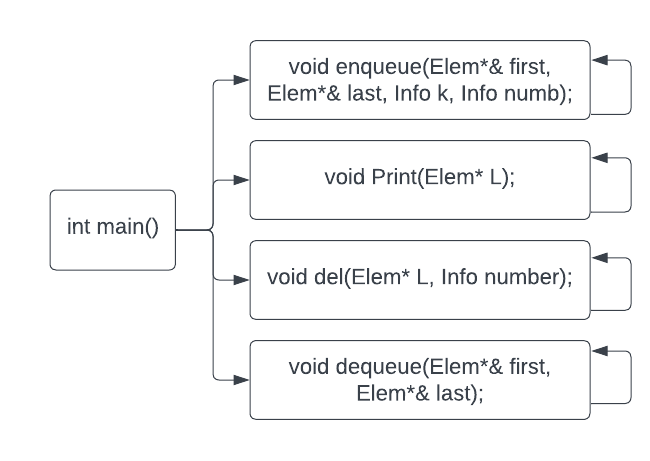
**студента групи ІН-105Б**

**Горанова Анастаса**

**Умова завдання:**



**Структурна схема:**



**Текст програми:**

// 12.2REC.cpp

// < Горанов Анастас >

// Лабораторна робота № 12.2REC

// Опрацювання лінійного однонаправленого списку

// Варіант 10

#include <iostream>

using namespace std;

typedef int Info; // створення псевдоніму типу

struct Elem

{

Elem\* link;

Info info;

};

void enqueue(Elem\*& first, Elem\*& last, Info k, Info numb);

void Print(Elem\* L);

void del(Elem\* L, Info number);

void dequeue(Elem\*& first, Elem\*& last);

int main()

{

Elem\* first = NULL,

\* last = NULL;

enqueue(first, last, 1, 15);

Print(first); // Вивід

cout << endl;

Info number;

cout << "Enter a number after which all items will be deleted: "; cin >> number;

del(first, number); // видалення компонентів

dequeue(first, last); // Вивід

cout << endl;

return 0;

}

void enqueue(Elem\*& first, Elem\*& last, Info k, Info numb)

{

Elem\* tmp = new Elem;

tmp->info = k;

tmp->link = NULL;

if (last != NULL)

last->link = tmp;

last = tmp;

if (first == NULL)

first = tmp;

if (k < numb) {

k++;

enqueue(first, last, k, numb);

}

}

void dequeue(Elem\*& first, Elem\*& last)

{

Elem\* tmp = first->link;

Info value = first->info;

delete first;

first = tmp;

if (first == NULL)

last = NULL;

cout << value << " ";

if (last != NULL) {

dequeue(first, last);

}

}

void Print(Elem\* L)

{

if (L != NULL) {

cout << L->info << " ";

L = L->link;

Print(L);

}

}

void del(Elem\* L, Info number)

{

if (L != NULL && L->link != NULL) {

if (L->link->info > number) {

Elem\* tmp = L->link->link;

delete L->link;

L->link = tmp;

del(L, number);

}

else

del(L->link, number);

}

}

**Посилання на git-репозиторій з проектом:** *https://github.com/StassNG/12.2REC*

**Результати програми та unit-тесту:**

****

**Висновок:** Я навчився опрацьовувати лінійні однонаправлені списки.