Національний технічний університет України

«Київський політехнічний інститут»

Кафедра автоматизованих систем обробки інформації та управління

Звіт про виконання лабораторної роботи №6 з дисципліни

«Алгоритми та структури даних»

Перевірено: Ковалюк Т.В. Виконав ст. 1 курсу ФОІТ

Проскура С.Л. гр. ІС-52

Дорошенко А.В.

Київ 2016

## Лабораторна робота 5

# **Динамічні структури. Лінійні списки**

Варіант № 9

*Мета:*

1. Вивчити особливості використання  динамічних структур типу списків.
2. Навчитися застосовувати списків в практичних задачах.

### Завдання

|  |  |
| --- | --- |
| Створити кільцевий список, елементами якого є числа. Послідовно вилучати кожне третє число. Підрахувати кількість вилучених чисел. Вивести початковий список та елементи, що вилучаються. |  |

### Блок-схема алгоритму



Рис.1 (Блок-схема головного алгоритму)



Рис.2 (Блок-схема алгоритму створення кільцевого списку)



Рис.3 (Блок-схема копіювання черги та виведення результатів)



Рис.4 (Блок-схема алгоритму видалення кожного третього елемента та його виведення)

### Код програми

//Laboratory work

//written by student of the first curse

//of the group IC-52 Anton Doroshenko

//2016.04.24

//==========================================================================

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include<iostream>

#include<conio.h>

#include<iomanip>

using namespace std;

struct Item

{

int val;//значення

Item\* next;//покажчик на наступний елемент

};

Item\* newptr = NULL;

Item\* previous;

Item\* head = NULL;//голова

Item\* cur;//поточний елемент

Item\* last = NULL;//кінець

int N = 0;//кількість елементів

//======= Сворення кыльцевого списку ===========

void create()

{

cout << "Enter number of components" << endl;

cin >> N;

cout << "Enter value" << endl;

for (int i = 0; i < N; i++)

{

newptr = new Item;

cin >> newptr->val;

if (head == NULL)

{

head = newptr;

last = head;

head->next = last;

last->next = head;

}

else

{

newptr->next = head;

last->next = newptr;

last = newptr;

}

}

}

//======= Виведення списку ===========

void output()

{

cur = head;

if (head == NULL)

{

cout << "List is empty" << endl;

}

else

{

cout << "List: ";

while (true)

{

cout << cur->val << " ";

cur = cur->next;

if (cur == head)

{

break;

}

}

cout << endl;

}

}

//===== Видалення кожного 3 елемента списку та його виведення ========

void del(Item\* ptr)

{

/\*int m = 0;\*/

cur = ptr;

for (int i = 0; i < 2; i++)

{

cur = cur->next;

if (i == 0)

{

previous = cur;

}

}

previous->next = cur->next;

cout << cur->val << endl;

cur = NULL;

/\*m++;\*/

if (previous == previous->next)

{

cout << previous->val << endl;

delete previous;

}

else/\* if (m < N)\*/

{

del(previous->next);

}

}

int main()

{

cout << "laboratory work number 6 made by Anton Doroshenko, IS-52" << endl;

cout << endl;

create();

output();

if (head != 0)

{

cout << "Elements which are deleting: " << endl;

del(head);

}

system("pause");

}

1. **Screen Shot результатів**

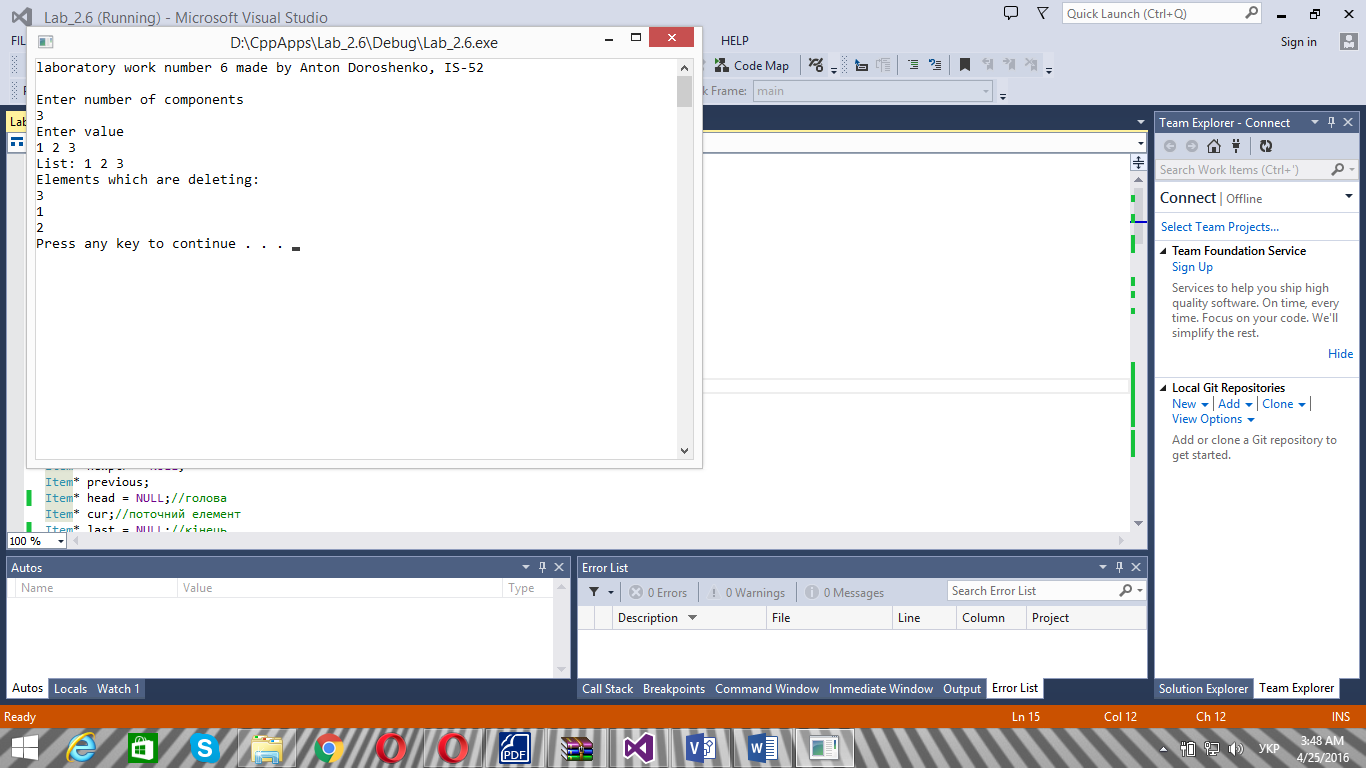


Рис.5 (Screen Shot результатів 1)

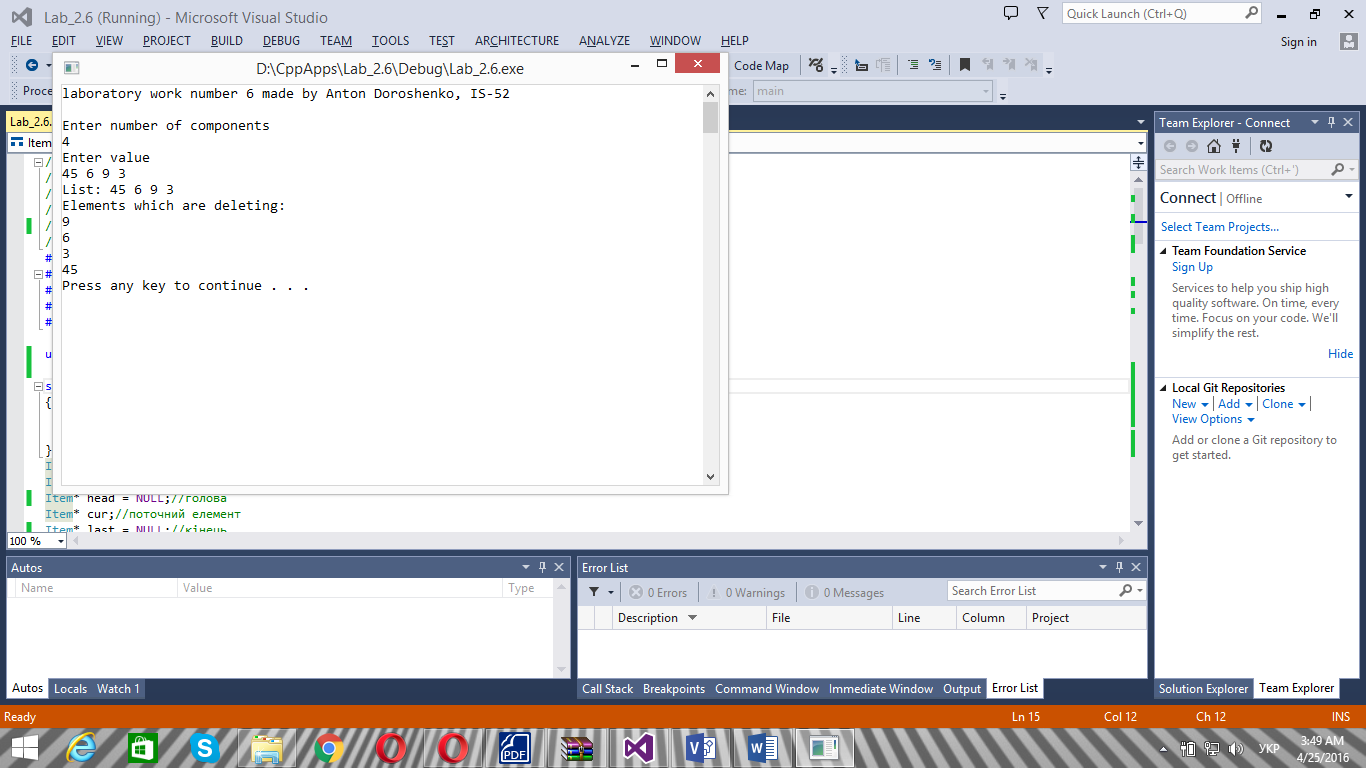


Рис.6 (Screen Shot результатів 2)

1. **Аналіз результатів**

Як ми бачимо зі ScreenShot’ів

Програма дає змогу користувачу створити кільцевий список. Якщо список не пустий програма видаляє та виводить на консоль кожний третій елемент.

1. **Висновок**

Програма працює правильно, про що свідчить аналіз результатів та ScreenShot’и. Програма коректно виводить результати. Програма застосовує динамічну структуру даних члінійний список.