Федеральное государственное автономное образовательное учреждение высшего образования   
«Пермский национальный исследовательский политехнический университет»

Лабораторная работа  
«Динамические структуры»

Выполнил студент группы ИВТ-23-2Б

Муравьев Дмитрий Александрович  
Проверила: доцент кафедры ИТАС

Ольга Андреева Полякова

2023

1. Постановка задачи:

Записи в линейном списке содержат ключевое поле типа \*char(строка символов). Сформировать двунаправленный список. Удалить элемент с заданным ключом. Добавить К элементов перед элементом с заданным номером.

2. Анализ задачи:

1. Создадим структуру элемента списка, где помимо указателей на предыдущий и следующий элементы будет храниться ключ в виде строки символов.

2. Реализуем функцию для удаления элемента с заданным ключом из двусвязанного списка:

- Обойдем список и найдем элемент с заданным ключом.

- Создадим временные указатели на предыдущий и следующий элементы.

- Перенаправим указатели, чтобы исключить из списка элемент с заданным ключом.

- Удалим элемент.

3. Напишем функцию для добавления К элементов перед элементом с заданным номером:

- Обойдем список до нужного элемента.

- Создадим новые элементы и свяжем их между собой и с существующими элементами.

3. Блок схема

4. Код программы

#include <iostream>

#include <string>

using namespace std;

template<typename T>

class List {

public:

void pop\_front();

T& GetValue(const int index);

void push\_front(T data);

void insert(T data, int index);

void removeIndex(int index);

private:

template<typename T>

class Node {

public:

Node\* pNext;

T data;

Node(T data = T(), Node\* pNext = nullptr) {

this->data = data;

this->pNext = pNext;

}

};

int Size;

Node<T>\* head;

};

int main() {

setlocale(LC\_ALL, "ru");

system("chcp 1251>NULL");

List<string> list;

cout << "Введите размер списка: ";

int size;

cin >> size;

string input;

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

getline(cin, input);

list.push\_front(input);

}

cout << endl;

cout << "Изначальное содержимое списка: " << endl;

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

cout << list.GetValue(i) << endl;

}

cout << "Введите ключ, для удаления элемента: " << endl;

getline(cin, input);

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

if (list.GetValue(i) == input) {

list.removeIndex(i);

}

}

cout << "Содержимое списка после удаления элемента с ключом \"" << input << "\":" << endl;

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

cout << list.GetValue(i) << endl;

}

cout << "Введите номер элемента, перед которым будем вставлять новые элементы: " << endl;

int number;

cin >> number;

cout << "Введите сколько элементов нужно добовить: " << endl;

int count;

cin >> count;

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

getline(cin, input);

list.insert(input, number++);

}

cout << "Список после вставки новых элементов: " << endl;

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

if (list.GetValue(i) != "")

cout << list.GetValue(i) << endl;

}

return 0;

}

template<typename T>

T& List<T>::GetValue(const int index) {

int counter = 0;

Node<T>\* current = this->head;

while (current != nullptr) {

if (counter == index) {

return current->data;

}

current = current->pNext;

counter++;

}

}

template<typename T>

void List<T>::push\_front(T data) {

head = new Node<T>(data, head);

Size++;

}

template<typename T>

void List<T>::insert(T data, int index) {

if (index == 0) {

push\_front(data);

}

else {

Node<T>\* previous = this->head;

for (int i = 0; i < index - 1; i++) {

previous = previous->pNext;

}

Node<T>\* newNode = new Node<T>(data, previous->pNext);

previous->pNext = newNode;

Size++;

}

}

template<typename T>

void List<T>::pop\_front() {

Node<T>\* temp = head;

head = head->pNext;

delete temp;

Size--;

}

template<typename T>

void List<T>::removeIndex(int index) {

if (index == 0) {

pop\_front();

}

else {

Node<T>\* previous = this->head;

for (int i = 0; i < index - 1; i++) {

previous = previous->pNext;

}

Node<T>\* toDelete = previous->pNext;

previous->pNext = toDelete->pNext;

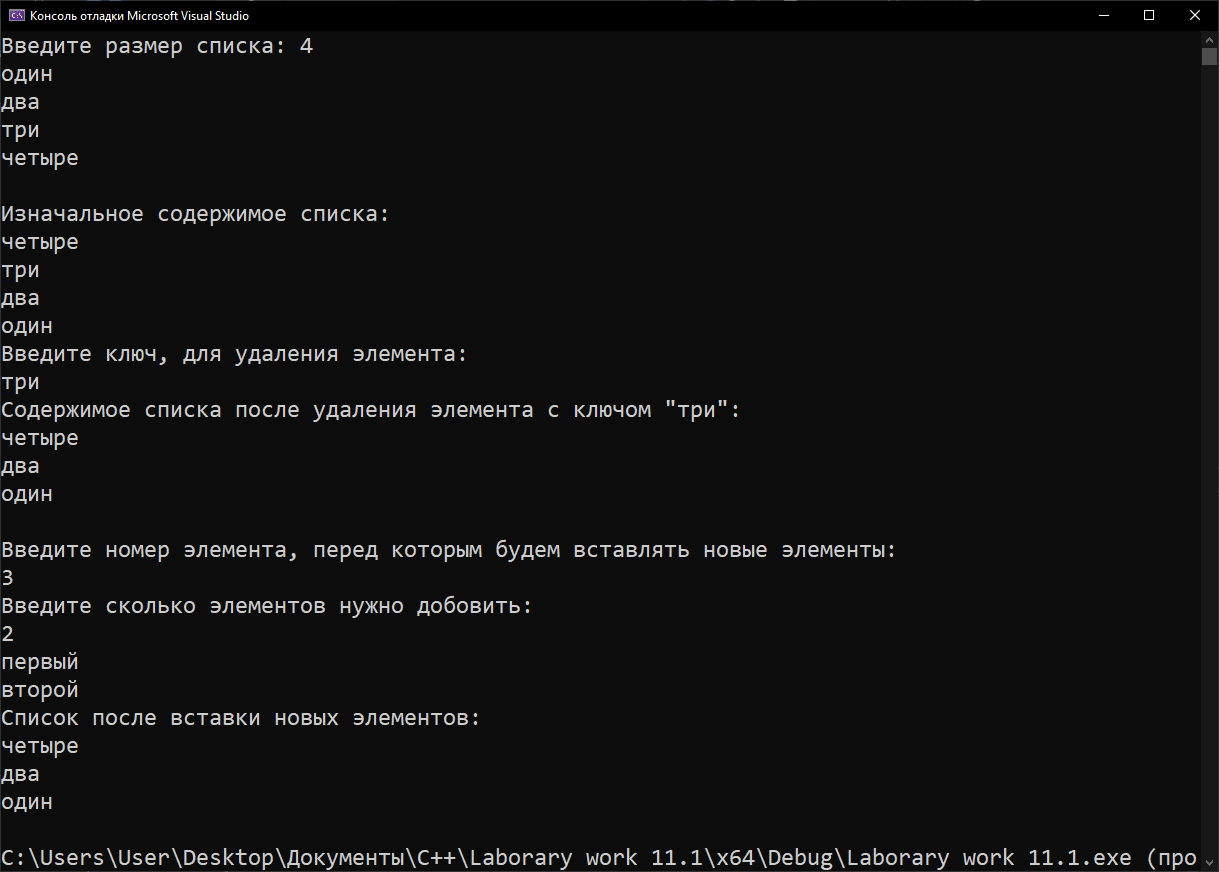
delete toDelete;

Size--;

}

}

5. Вывод программы



6. Вывод

Мне удалось выполнить поставленное задание. В ходе выполнения задания я освоил новый приемы программирования.

7. Github