Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

Пермский национальный исследовательский политехнический университет

Электротехнический факультет

Кафедра информационных технологий и автоматизированных систем

**ОТЧЕТ**

**о работе по информатике**

Семестр: 2

На тему: «Информационные динамические структуры.

Однонаправленные списки»

Выполнил студент ИВТ-22-2б:

Захаров Дмитрий Сергеевич

Проверил доцент кафедры ИТАС:

Полякова Ольга Андреевна

Пермь 2023

**Постановка задачи**

Записи в линейном списке содержат ключевое поле типа int. Сформировать однонаправленный список. Удалить из него К элементов, начиная с заданного номера, добавить элемент перед элементом с заданным ключом

**Алгоритм решения**

1. Создать функцию создания списка
2. Создать функцию добавления элемента
3. Создать функцию удаления элемента
4. Создать функцию выведения списка на экран
5. Создать функцию записи списка в файл
6. Создать функцию удаления списка
7. Создать функцию чтения списка из файла
8. Создать главную функцию main

**Блок-схема**

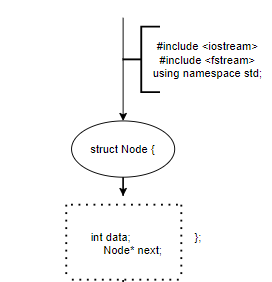
****

Рис. 1 – Блок-схема структуры Node

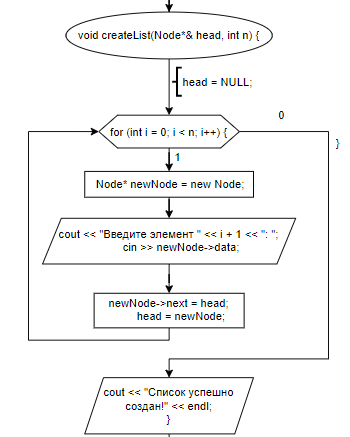
****

Рис. 2 – Блок-схема функции void createList

****

Рис. 3 – Блок-схема функции void print\_list

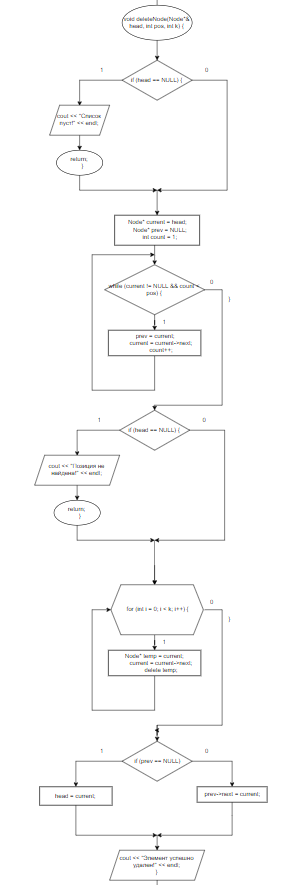
****

Рис. 4 – Блок-схема функции void deleteNode

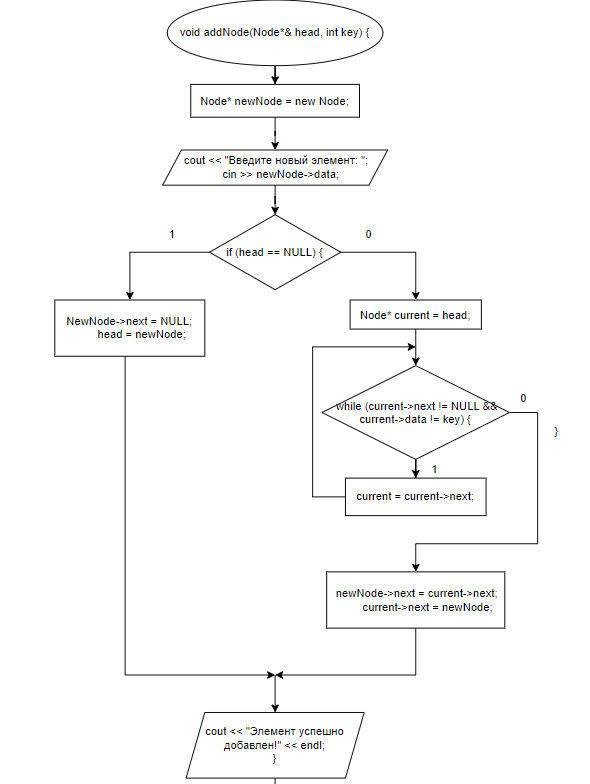
****

Рис. 5 – Блок-схема функции void Addnode()

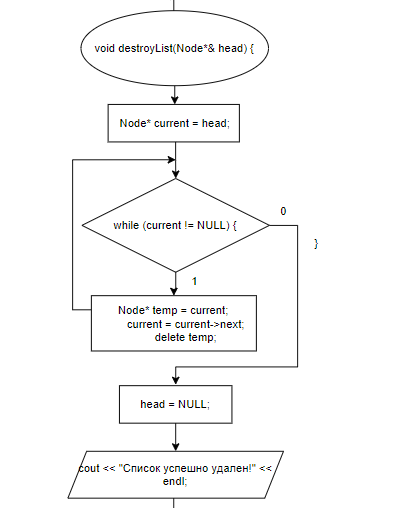
****

Рис. 6 – Блок-схема функции void destroyList

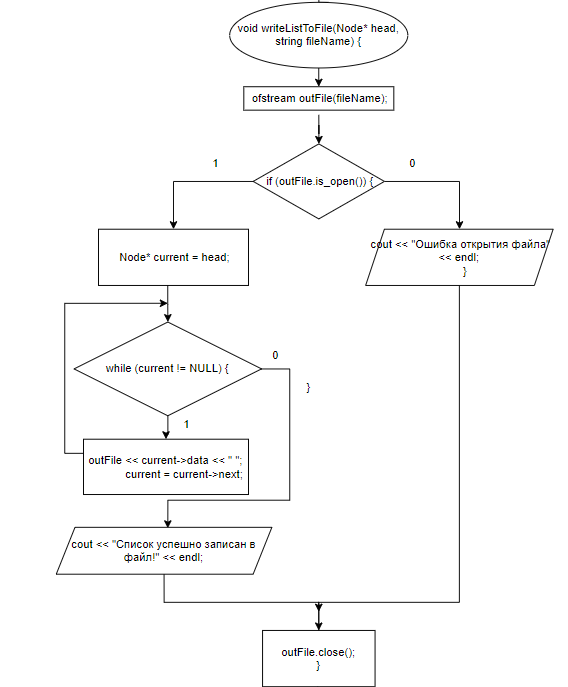
****

Рис. 7 – Блок-схема функции void writeListToFile

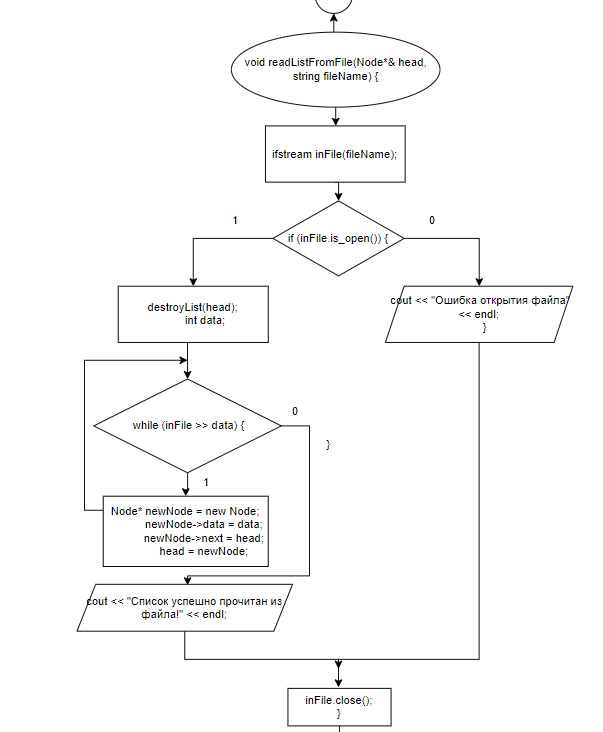
****

Рис. 8 – Блок-схема функции void readListFromFile

**Код программы**

#include <iostream>

#include <fstream>

using namespace std;

struct Node {

int data;

Node\* next;

};

void createList(Node\*& head, int n) {

head = NULL;

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

Node\* newNode = new Node;

cout << "Введите элемент " << i + 1 << ": ";

cin >> newNode->data;

newNode->next = head;

head = newNode;

}

cout << "Список успешно создан!" << endl;

}

void addNode(Node\*& head, int key) {

Node\* newNode = new Node;

cout << "Введите новый элемент: ";

cin >> newNode->data;

if (head == NULL) {

newNode->next = NULL;

head = newNode;

}

else {

Node\* current = head;

while (current->next != NULL && current->data != key) {

current = current->next;

}

newNode->next = current->next;

current->next = newNode;

}

cout << "Элемент успешно добавлен!" << endl;

}

void pop(Node\*& head, int k) {

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

Node\* temp = head;

head = head->next;

delete temp;

}

}

void printList(Node\* head) {

cout << "Содержание списка: ";

Node\* current = head;

while (current != NULL) {

cout << current->data << " ";

current = current->next;

}

cout << endl;

}

void writeListToFile(Node\* head, string fileName) {

ofstream outFile(fileName);

if (outFile.is\_open()) {

Node\* current = head;

while (current != NULL) {

outFile << current->data << " ";

current = current->next;

}

cout << "Список успешно записан в файл!" << endl;

}

else {

cout << "Ошибка открытия файла" << endl;

}

outFile.close();

}

void destroyList(Node\*& head) {

Node\* current = head;

while (current != NULL) {

Node\* temp = current;

current = current->next;

delete temp;

}

head = NULL;

cout << "Список успешно удален!" << endl;

}

void readListFromFile(Node\*& head, string fileName) {

ifstream inFile(fileName);

if (inFile.is\_open()) {

destroyList(head);

int data;

while (inFile >> data) {

Node\* newNode = new Node;

newNode->data = data;

newNode->next = head;

head = newNode;

}

cout << "Список успешно прочитан из файла!" << endl;

}

else {

cout << "Ошибка открытия файла" << endl;

}

inFile.close();

}

int main() {

setlocale(0, "RUS");

Node\* head = NULL;

int n, key, k;

string fileName;

cout << "Введите количество элементов списка: ";

cin >> n;

createList(head, n);

printList(head);

cout << "Введите ключ для добавления элемента перед: ";

cin >> key;

addNode(head, key);

printList(head);

cout << "Введите количество элементов, которое удалится с конца: ";

cin >> k;

pop(head, k);

printList(head);

cout << "Введите имя файла для записи списка: ";

cin >> fileName;

writeListToFile(head, fileName);

destroyList(head);

cout << "Введите имя файла для чтения списка: ";

cin >> fileName;

readListFromFile(head, fileName);

printList(head);

return 0;

}