Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное учреждение высшего образования «Пермский национальный исследовательский политехнический университет»

Электротехнический факультет Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т по лабораторной работе**

Дисциплина: «Основы алгоритмизации и программирования»

Выполнил студент группы ИВТ21-1Б Лебедев Т.А.

Проверил Яруллин Д.В.

Пермь, 2021

**Постановка задачи:**

1. Сформировать однонаправленный и двунаправленный списки или стек и очередь. Тип информационного поля указан в варианте.

2. Распечатать полученную структуру.

3. Выполнить обработку структуры в соответствии с заданием.

4. Распечатать полученный результат.

5. Удалить соответствующую структуру из памяти.

**Анализ решения задачи:**

структура с названием Node

информационное поле типа char

указатель на следующий элемент

указатель на предыдущий элемент

функция для создания очереди

выделяем память для нового элемента очереди

просим ввести данные

проверяем первый ли элемент очереди

функция для вывода очереди

подключаем русскую раскладку

просим ввести размер очереди

вводим номер нового элемента

добавляем элемент

выводим очередь

**Описание переменных:**

n- количество элементов стека

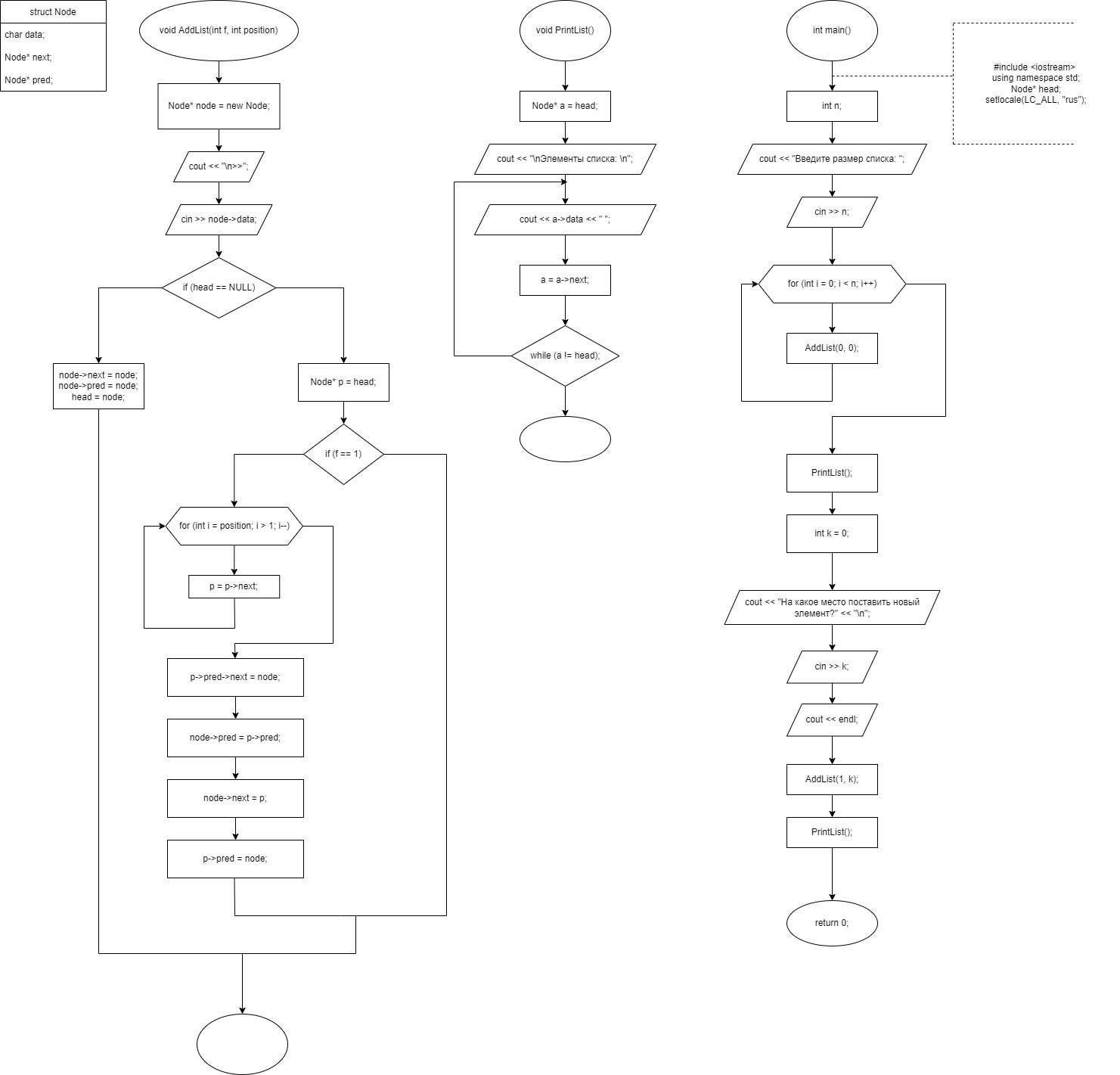
data-информационное поле структуры

\*next-указатель на следующий элемент

\*first-переменная, в которой будет хранится первый элемент списка

\*p,\*r-вспомогательные переменные

**Блок схема:**

****

**Код программы:**

#include <iostream>

using namespace std;

struct Node

{

char data;

Node\* next;

Node\* pred;

};

Node\* head;

void AddList(int f, int position)

{

Node\* node = new Node;

cout << "\n>>";

cin >> node->data;

if (head == NULL)

{

node->next = node;

node->pred = node;

head = node;

}

else

{

Node\* p = head;

if (f == 1)

{

for (int i = position; i > 1; i--)

p = p->next;

}

p->pred->next = node;

node->pred = p->pred;

node->next = p;

p->pred = node;

}

}

void PrintList()

{

Node\* a = head;

cout << "\nЭлементы очереди: \n";

do

{

cout << a->data << " ";

a = a->next;

} while (a != head);

}

int main()

{

setlocale(LC\_ALL, "rus");

int n;

cout << "Введите размер очереди: ";

cin >> n;

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

{

AddList(0, 0);

}

PrintList();

int k = 0;

cout << "На какое место поставить новый элемент?" << "\n";

cin >> k;

cout << endl;

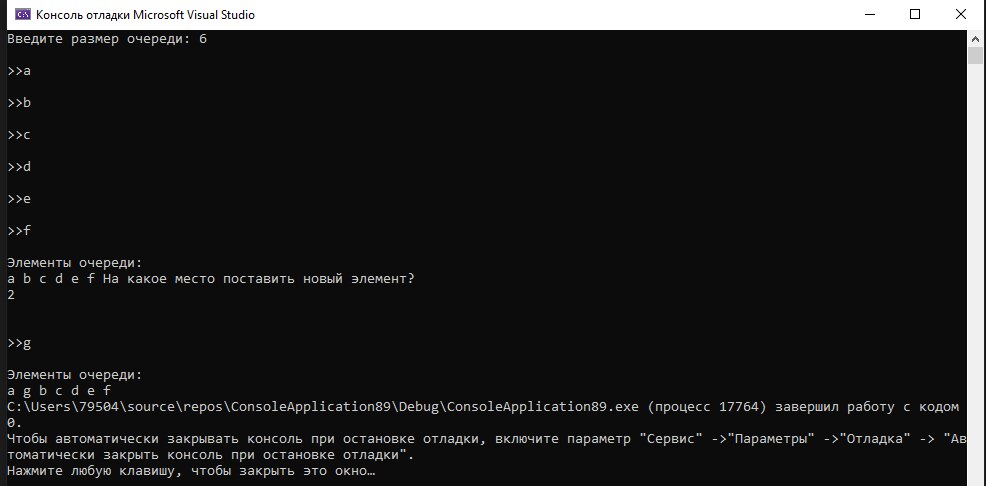
AddList(1, k);

PrintList();

return 0;

}

**Скриншоты результатов:**

****