Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики

и радиоэлектроники»

Специальность «Программная инженерия»

Кафедра инженерной психологии и эргономики

Учебная дисциплина «Основы алгоритмизации и программирования»

ОТЧЕТ

по лабораторной работе №11

«Списки. Стек»

Вариант 14

Подготовил: Зайцев Е. А.

Проверил: Усенко Ф.В.

Минск 2025

**Цель работы:** сформировать умения и навыки написания программ с использованием стека, списков.

**Задание:** Создать список из случайных целых чисел. Перенести в другой список все элементы, находящиеся между вершиной и элементом с минимальным значением.

Код программы приведен ниже:

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

struct Node {

int data;

Node\* next;

};

int main() {

srand(time(0)); //Инициализация генератора случайных чисел

//Создаем список из 10 случайных чисел (от 0 до 99)

Node\* head = nullptr;

Node\* tail = nullptr;

for (int i = 0; i < 10; i++) {

Node\* newNode = new Node;

newNode->data = rand() % 100;

newNode->next = nullptr;

if (head == nullptr) {

head = newNode;

tail = newNode;

}

else {

tail->next = newNode;

tail = newNode;

}

}

//Выводим исходный список

cout << "Original list: ";

Node\* current = head;

while (current != nullptr) {

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

current = current->next;

}

cout << endl;

//Находим узел с минимальным значением

Node\* minNode = head;

Node\* temp = head->next;

while (temp != nullptr) {

if (temp->data < minNode->data) {

minNode = temp;

}

temp = temp->next;

}

cout << "Min value: " << minNode->data << endl;

//Создаем новый список для элементов между головой и minNode

Node\* newListHead = nullptr;

Node\* newListTail = nullptr;

current = head->next; // Начинаем с элемента после головы

while (current != minNode && current != nullptr) {

// Создаем копию текущего узла для нового списка

Node\* newNode = new Node;

newNode->data = current->data;

newNode->next = nullptr;

if (newListHead == nullptr) {

newListHead = newNode;

newListTail = newNode;

}

else {

newListTail->next = newNode;

newListTail = newNode;

}

current = current->next;

}

//Выводим новый список

cout << "Elements between head and min: ";

current = newListHead;

while (current != nullptr) {

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

current = current->next;

}

cout << endl;

return 0;

}

Результат работающей программы представлен на рисунке 1:

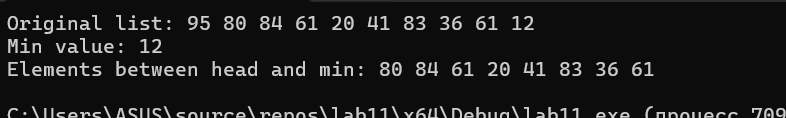


Рисунок 1 – Результат работы программы

Блок-схема работы программы представлена на рисунке 2.

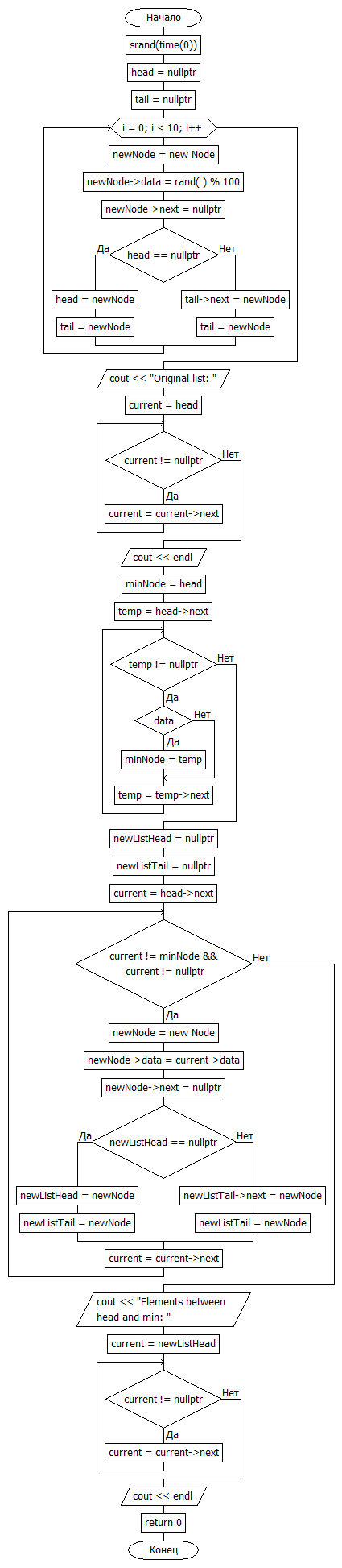


Рисунок 2

**Вывод:** в ходе выполнения работы была достигнута цель данной лабораторной работы: сформировать умения и навыки написания программ с использованием стека, списков.