МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

##### «МИНСКИЙ ГОСУДАРСТВЕННЫЙ КОЛЛЕДЖ ЦИФРОВЫХ ТЕХНОЛОГИЙ»

##### ***Группа 82 ТП***

## **Отчет**

## **по лабораторной работе № 27**

# " Обработка списков"

учебный предмет

«Основы алгоритмизации и программирования»

**Исполнитель: Шатуха А.К.**

**Руководитель: Д.А.Федкевич**

Минск 2025

# ЗАДАНИЕ

### Задание 1. Однонаправленные списки

Написать программу по созданию, просмотру, добавлению и решению поставленной задачи для однонаправленного линейного списка (стек и/или очередь).

1. Создать список из случайных чисел, вычислить среднее арифметическое и заменить им первый элемент.

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

struct Item {

int info;

Item\* next;

};

class Stack {

private:

Item\* first;

public:

Stack() : first(NULL) {}

~Stack() {

while (first) {

Item\* temp = first;

first = first->next;

delete temp;

}

}

void push(int value) {

Item\* newItem = new Item;

newItem->info = value;

newItem->next = first;

first = newItem;

}

void createRandomList(int n) {

srand(time(0));

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

push(rand() % 100);

}

}

void show() {

Item\* current = first;

while (current) {

cout << current->info << " ";

current = current->next;

}

cout << endl;

}

void replaceFirstWithAverage() {

if (!first) return;

Item\* current = first;

double sum = 0;

int count = 0;

while (current) {

sum += current->info;

count++;

current = current->next;

}

double average = sum / count;

first->info = static\_cast<int>(average);

}

};

int main() {

setlocale(LC\_ALL, "ru");

Stack stack;

int n;

cout << "Введите количество элементов в списке: ";

cin >> n;

stack.createRandomList(n);

cout << "Исходный список: ";

stack.show();

stack.replaceFirstWithAverage();

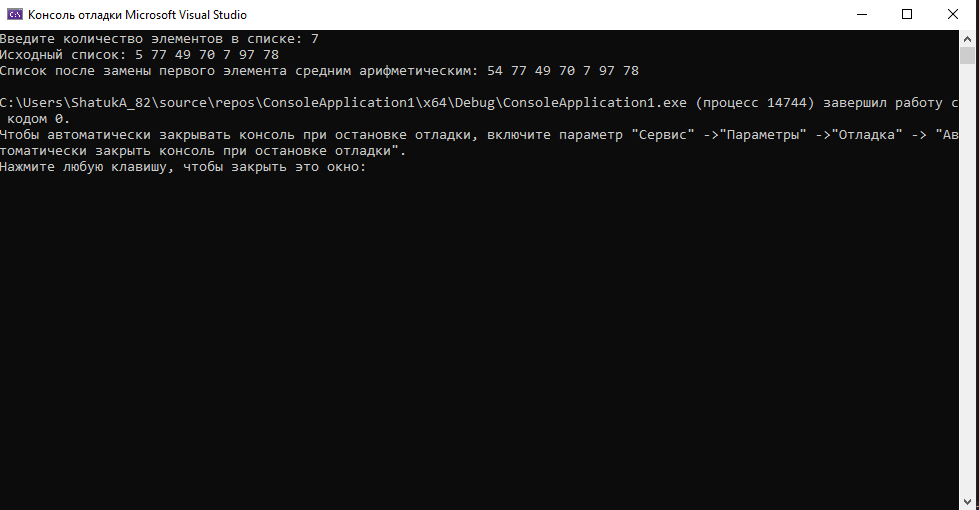
cout << "Список после замены первого элемента средним арифметическим: ";

stack.show();

return 0;

}

Скриншот:



### Задание 2. Двунаправленные списки

Написать программу по созданию, просмотру, добавлению и решению поставленной задачи для двунаправленного линейного списка.

1. Создать список из случайных целых чисел. Удалить из списка элементы с повторяющимися более одного раза значениями.

Листинг или ссылка на проект:

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

struct Node {

int data;

Node\* next;

Node\* prev;

};

class DoublyLinkedList {

private:

Node\* head;

Node\* tail;

public:

DoublyLinkedList() : head(NULL), tail(NULL) {}

~DoublyLinkedList() {

while (head) {

Node\* temp = head;

head = head->next;

delete temp;

}

}

void append(int value) {

Node\* newNode = new Node;

newNode->data = value;

newNode->next = NULL;

newNode->prev = tail;

if (!head) {

head = tail = newNode;

}

else {

tail->next = newNode;

tail = newNode;

}

}

void createRandomList(int n) {

srand(time(0));

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

append(rand() % 20);

}

}

void display() {

Node\* current = head;

while (current) {

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

current = current->next;

}

cout << endl;

}

void removeDuplicates() {

if (!head || !head->next) return;

Node\* current = head;

while (current) {

Node\* runner = current->next;

while (runner) {

if (runner->data == current->data) {

Node\* temp = runner;

runner = runner->next;

if (temp->prev) {

temp->prev->next = temp->next;

}

if (temp->next) {

temp->next->prev = temp->prev;

}

if (temp == tail) {

tail = temp->prev;

}

delete temp;

}

else {

runner = runner->next;

}

}

current = current->next;

}

}

};

int main() {

setlocale(LC\_ALL, "ru");

DoublyLinkedList list;

int n;

cout << "Введите количество элементов в списке: ";

cin >> n;

list.createRandomList(n);

cout << "Исходный список: ";

list.display();

list.removeDuplicates();

cout << "Список после удаления повторяющихся элементов: ";

list.display();

return 0;

}

Скриншот:

