

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

ЭЛЕКТРОТЕХНИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА «ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
АВТОМАТИЗИРОВАННЫХ СИСТЕМ»

ОТЧЁТ

«ЛАБОРАТОРНАЯ №6:

КЛАССЫ — КОНТЕЙНЕРЫ И ИТЕРАТОРЫ»

Дисциплина: «Программирование»

Выполнил:

Студент группы ИВТ-21-26

Безух Владимир Сергеевич

Проверил:

Доцент кафедры ИТАС

Полякова Ольга Андреевна

Пермь, 2022

Содержание

1.	Постановка задачи	3
2.	Контрольные вопросы	4
3.	Анализ результатов.....	6

1. Постановка задачи

1. Класс-контейнер SinglyList<T> типа int. Реализовать операции: доступа по индексу [], определение размера списка size(), перегрузка оператора * для произведения всех значений списка на N, сдвиг итератора на + N позиций.

2. Контрольные вопросы

1. Что такое абстрактный тип данных? Привести примеры АДТ.

АДТ — тип данных, определяемый только через операции, которые могут выполняться над соответствующими объектами безотносительно к способу представления этих объектов.

2. Привести примеры абстракции через параметризацию.

Функция с параметрами, когда мы не задумываемся о её реализации.

3. Привести примеры абстракции через спецификацию.

Функция без параметров, когда мы не задумываемся о её реализации.

4. Что такое контейнер? Привести примеры.

Набор однотипных элементов. Например, в STL много разных контейнеров: vector, list, map и т.д..

5. Какие группы операций выделяют в контейнерах?

Операции доступа к элементам (и замены элементов), операции добавления и удаления элементов или групп элементов, операции поиска элементов и групп элементов, объединения контейнеров, специальные операции, уникальные для конкретного контейнера.

6. Какие виды доступа к элементам контейнера существуют? Привести примеры.

Последовательный, прямой и ассоциативный. Прямой — доступ по индексу. Например, через оператор []. Ассоциативный по принципу «ключ-значение». Последовательный через итераторы.

7. Что такое итератор?

Объект, обеспечивающий последовательный доступ к элементам контейнера.

8. Каким образом может быть реализован итератор?

Как класс с методами, являющийся частью класса-контейнера.

9. Каким образом можно организовать объединение контейнеров?

Простое объединение, объединение упорядоченных контейнеров, объединение множеств (уникальные элементы), пересечение множеств (есть в обоих контейнерах), через вычитание множества, через извлечение части и создание нового.

10. Какой доступ к элементам предоставляет контейнер, состоящий из элементов «ключ-значение»?

Ассоциативный доступ.

11. Как называется контейнер, в котором вставка и удаление элементов выполняется на одном конце контейнера?

Стек.

12. Контейнер реализован как динамический массив, в нем определена операция доступа по индексу. Каким будет доступ к элементам контейнера?

Прямой доступ.

13. Контейнер реализован как линейный список. Каким будет доступ к элементам контейнера?

Последовательный доступ.

3. Анализ результатов

Результаты выполнения программы (рис. 1).

```
#include "SinglyList.h"

int main()
{
    SinglyList<int> list;

    list.pushBack(0); list.pushBack(1); list.pushBack(2); list.pushBack(3);
    for (const int& value : list)
        std::cout << value << " ";
    std::cout << std::endl;

    // Доступ к элементам через []
    list[0] = 42;
    for (const int& value : list)
        std::cout << value << " ";
    std::cout << std::endl;

    // Размер списка
    std::cout << list.size() << std::endl;

    // Умножение всех элементов списка на целочисленное число
    list * 10;
    for (const int& value : list)
        std::cout << value << " ";
    std::cout << std::endl;

    // Сдвиг итератора на N элементов вправо
    auto iter = list.begin();
    std::cout << *iter << " ";
    std::cout << *(iter + 2);
}
```

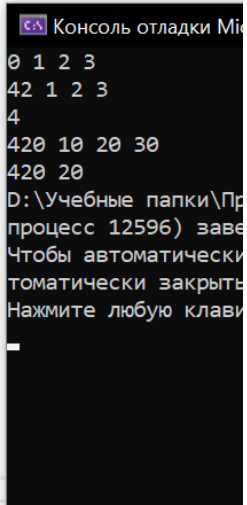


Рисунок 1 — Результаты