МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Нижегородский государственный университет им. Н.И. Лобачевского»**

**Национальный исследовательский университет**

**Институт информационных технологий, математики и механики**

**Кафедра математического обеспечения и суперкомпьютерных технологий**

**ЛАБОРАТОРНАЯ РАБОТА**

**«Список на массиве»**

**Выполнил:**студент группы 381706-2

Гущин Александр Владимирович

\_\_\_\_\_\_\_\_\_\_\_  Подпись

**Руководитель:**

Ассистент кафедры МОСТ

Лебедев Илья Геннадьевич

\_\_\_\_\_\_\_\_\_\_\_  Подпись

Нижний Новгород

2018

Оглавление

[1.Введение 3](#_Toc5832442)

[2.Постановка задачи 4](#_Toc5832443)

[3.Руководство пользователя 5](#_Toc5832444)

[4.Руководство программиста 6](#_Toc5832445)

[4.1.Описание структуры программы 6](#_Toc5832446)

[4.2.Описание структур данных 6](#_Toc5832447)

[4.3.Описание алгоритмов 6](#_Toc5832448)

[5.Заключение 8](#_Toc5832449)

[6.Литература 9](#_Toc5832450)

1.Введение

Список на массиве - базовая структура данных в информатике, состоящая из массива элементов и массива указателей. Принципиальным преимуществом перед массивом является структурная гибкость: порядок элементов связного списка может не совпадать с порядком расположения элементов данных в памяти компьютера, а порядок обхода списка всегда явно задаётся его внутренними связями.

2.Постановка задачи

Необходимо написать реализацию списка на массиве со всеми сопутствующими конструкторами, методами (положить элемент в начало, в конец, взять элемент из начало, конца).

3.Руководство пользователя

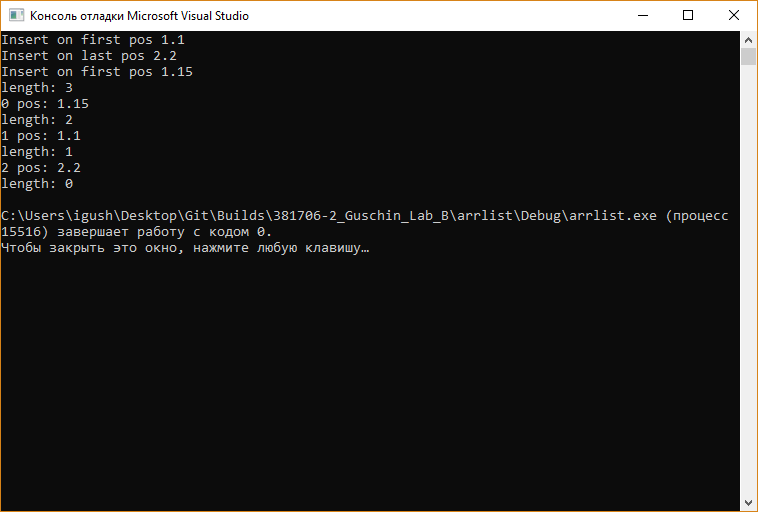


Рис.1 Пример использования.

Демонстрация методов класса(Создание, вставка в начало, в конец, взятие с удалением элемента).

4.Руководство программиста

4.1.Описание структуры программы

Программа состоит из таких модулей как:

ArrList.h – список.

4.2.Описание структур данных

1. T\* p – указатель на массив данных.
2. int\* ind – указатель на массив индексов.
3. int size – размер списка.
4. int count – кол-во элементов в списке.
5. int freeInd – индекс свободного элемента.
6. int firstInd – начальный индекс.
7. ArrList(const int \_size = 0) – конструктор с параметром.
8. ArrList(const ArrList& A) – конструктор копирования.
9. ~ArrList() – деструктор.
10. int GetLen() – метод, возвращает кол-во элементов.
11. bool IsFull() – метод, возвращает true если список заполнен.
12. bool IsEmpty() – метод, возвращает false если список пуст.
13. void InsFirst(const T& a) – метод, кладет элемент в начало списка.
14. void InsLast(const T& a) – метод, кладет элемент в начало списка.
15. T GetFirst() - метод, берет элемент с удалением из начала.
16. T GetLast() - метод, берет элемент с удалением из конца.

4.3.Описание алгоритмов

GetLast()

Если список пуст, то бросается исключение.

Если в списке только одно значение, то вызывается метод GetFirst().

Иначе создается две переменные типа int(prInd и resInd).

prInd = индексу первого элемента.

resInd = индексу второго элемента.

Пока индекс за следующий за resInd не будет обозначен как конечный, то prInd = равняется индексу следующего за ним элемент. Тоже самое с resInd.

После чего индекс после prInd обозначается как конечный.

Индекс после resInd обозначается как свободный.

Уменьшение счетчика элементов в списке на 1.

Возврат значения из массива p по индексу resInd.

5.Заключение

Благодаря этой работе у меня появился опыт взаимодействия со списком, улучшил понимание основ программирования базовых структур данных, осознал важность списка в жизни человека. Список значительно упрощает работу с информацией и помогают решать многие практические задания.

6.Литература

1. Википедия https://ru.wikipedia.org/wiki/Список\_(информатика) (Дата обращения 30.12.2018)
2. <https://rsdn.org/article/alg/list.xml> (Дата обращения 30.12.2018)