**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра информационных систем**

отчет

**по практической работе №2**

**по дисциплине «Программирование»**

Тема: **Строки. Символьные массивы**

|  |  |  |
| --- | --- | --- |
| Студент(ка) гр. |  | Власов И. Д |
| Преподаватель |  | Глущенко А. Г. |

Санкт-Петербург

2025

**Основные теоретические положения.**

## **Одномерные динамические массивы**

Для того чтобы создать в динамической области некоторый объект, необходима одна обычная переменная-указатель (не динамическая переменная). Сколько таких объектов понадобится для одновременной обработки, столько необходимо иметь обычных переменных-указателей. Таким образом, проблема задач неопределенной размерности созданием одиночных динамических объектов решена быть не может.

Решить эту проблему поможет возможность создавать в динамической области памяти массивы объектов с таким количеством элементов, которое необходимо в данный момент работы программы, т. е. создание динамических массивов. Действительно, для представления массива требуется всего одна переменная-указатель, а в самом массиве, на который ссылается этот указатель, может быть столько элементов, сколько требуется в данный момент времени.

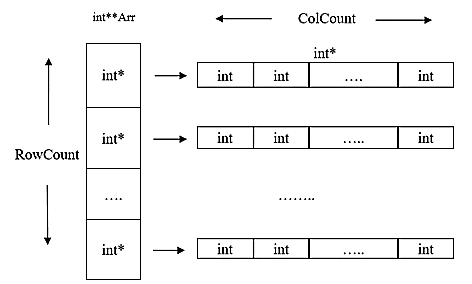
## **Изменение размера динамического массива**

Очень часто в процессе работы программы требуется изменять размеры уже созданных и заполненных данными массивов. Общий алгоритм решения этой задачи таков:

1.      создать исходный массив размерности n1 и заполнить его данными;  
2.      создать промежуточный массив размерности n2 (пусть n2 > n1);  
3.      скопировать данные из исходного массива в промежуточный массив;  
4.      освободить память от исходного массива;  
5.      переменной-указателю исходного массива присвоить значение переменной-указателя промежуточного массива;  
6.      заполнить новые элементы массива данными.

## Двумерные динамические массивы

Создание многомерных массивов несколько сложнее, чем одномерных динамических массивов.



Для того чтобы получить двумерный массив, необходимо:

1. создать одномерный динамический массив из **rowCount** указателей на базовый тип элементов массива (в нашем случае – указателей на тип **int**);  
   2)  в цикле создать **rowCount** одномерных динамических массивов, каждый из которых содержит **colCount** элементов базового типа (в нашем  случае –  указателей на тип **int**), и адреса их первых элементов записать в соответствующие элементы «вертикального» массива.

**Двусвязный список**

Каждый узел двусвязного (двунаправленного) линейного списка содержит два поля указателей – на следующий и на предыдущий узлы. Указатель на предыдущий узел корня списка содержит нулевое значение. Указатель последнего узла также содержит нулевое значение.

Поскольку каждый элемент списка должен иметь три части, логичнее всего представить его в виде следующей структуры:

struct List

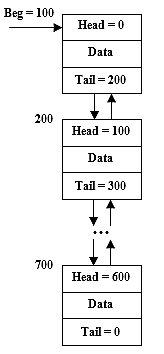
{

int data;

List \*head;

List \*tail;

};



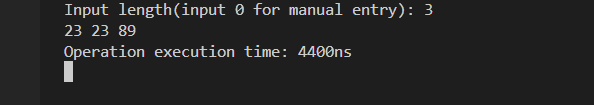
 Поле  **Head** содержит адрес предыдущего элемента, поле **Tail** содержит адрес следующего элемента списка. Такая организация списка позволяет перемещаться по его элементам в двух направлениях.

**Выполнение работы.**

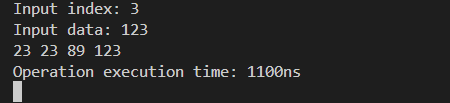
**1)**Меню действий



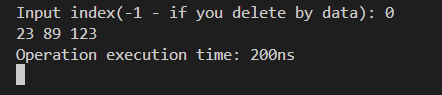
2)Создание списка/массива



3)Вставка



4)Удаление



5)Перестановка

