



## Постановка задачи

Разработать объект, который обладает следующей функциональностью:

- В конструкторе вводит количество элементов целочисленного массива, создает целочисленный массив заданным количеством элементов, определяет значения элементов массива. Значение каждого элемента равно квадрату индекса элемента.
- Выводит значения элементов массива в обратном порядке.

Необходимые свойства объекта определяется разработчиком. Количество элементов массива принадлежит интервалу от 5 до 20.

Написать программу:

1. Создает объект посредством оператора new.
2. Выводит значения элементов массива в обратном порядке.
3. Удаляет объект посредством оператора delete.

## Описание входных данных

Первая строка (количество элементов массива):  
«целое число в десятичном формате»

## Описание выходных данных

Первая строка, с первой позиции: целочисленные значения элементов массива в десятичном формате, каждое значение занимает 5 позиций

## Метод решения

Вспользуемся объектом q класса Arr

- класс Arr
  - Поля, отвечающие за количество элементов массива
  - Наименования - Arr
  - Тип - целочисленные

- Модификатор доступа - public

#### Методы

- Конструктор класса Arr() - вводит количество элементов целочисленного массива
- Метод squarCout() класса Arr() - выводит значения элементов массива в обратном порядке
- Деструктор ~Arr() класса Arr() - освобождение использованных объектов ресурсов и удаление нестатических переменных объекта

Для решения этой задачи воспользуемся объектами стандартных потоков ввода и вывода cin, cout, условным оператором if, манипулятора потока ввода/вывода setw, очищение выделенной памяти delete, оператора цикла со счетчиком из библиотеки <iostream>

### Описание алгоритма

Функция: main

Функционал: точка входа в программу

Параметры: -

Возвращаемое значение: int, 0

№	Предикат	Действия	№ перехода	Комментарий
1		Объявление i	2	Целочисленная переменная
2		Ввод целочисленной переменной i	3	
3	(x >=5 && x <=20)	Объявление объекта q класса Arr	3	
		Создаем экземпляр указателя на объект q	4	
4		Вызов метода squarCout	5	
5		Удаление объекта q	∅	Используем delete - удаление объекта

Класс объекта: Arr

Модификатор доступа: public

Метод: Agg

Функционал: Вводит количество элементов целочисленного массива

Параметры: x - целочисленная переменная

Возвращаемое значение: нет

№	Предикат	Действия	№ перехода	Комментарий
1		Присваивание целочисленной переменной i переменную x	2	
2		Выделение памяти динамического массива array под размер i	3	
3	$k < i$	присваиваем элементу массива с индексом k значение элемента в квадрате	3	
			∅	

Класс объекта: Agg

Модификатор доступа: public

Метод: squarCout

Функционал: Выводит значения элементов массива в обратном порядке

Параметры: -

Возвращаемое значение: нет

№	Предикат	Действия	№ перехода	Комментарий
1	$k \geq 0$	Уменьшение i на 1 Присваивание переменной k префиксный декремент	1	Цикл for перебирает элементы в обратном порядке, когда значение переменной k не станет равно 0 Используем функцию setw(5)
			∅	

Класс объекта: Agg

Модификатор доступа: public

Метод: ~Agg

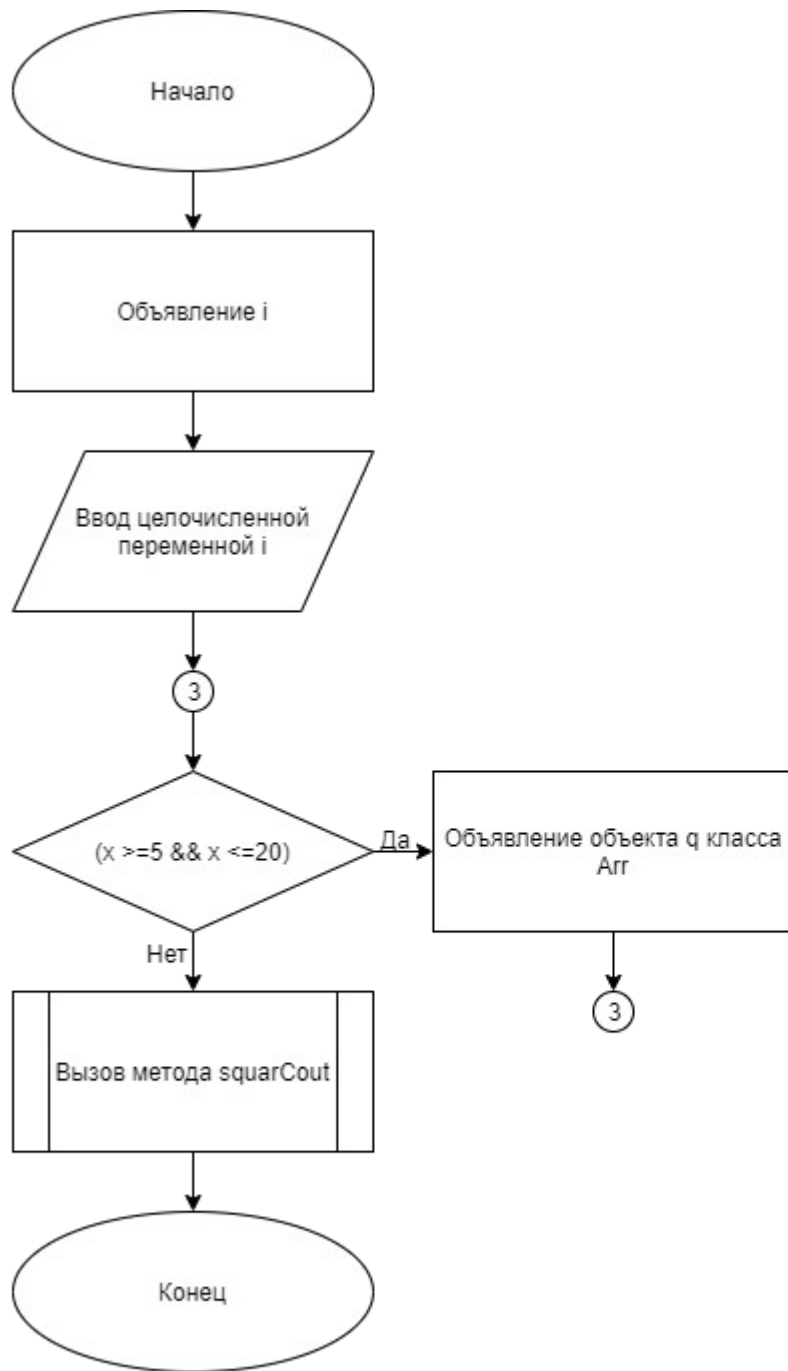
Функционал: Деструктор

класса Параметры: -

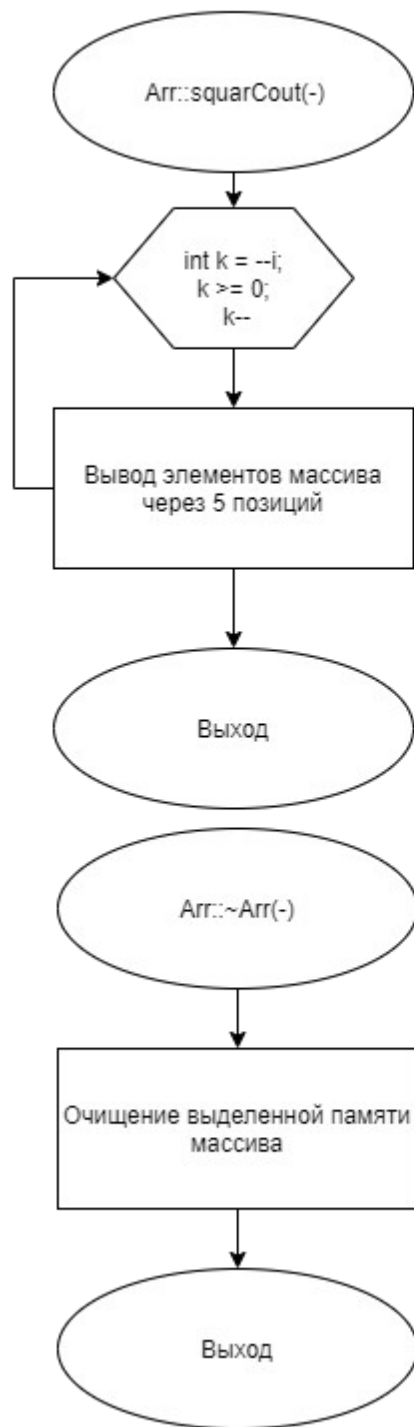
Возвращаемое значение: нет

<b>№</b>	<b>Предикат</b>	<b>Действия</b>	<b>№ перехода</b>	<b>Комментарий</b>
1		Очищение выделенной памяти массива	∅	

### **Блок-схема алгоритма**







**Код программы**

**Файл header.cpp**



```

#include "header.h"
using namespace std;
#include <iostream>
    Arr::Arr(int x) {
        i = x;
        array = new int[i];
        for(int k = 0; k < i; k++) {
            array[k] = k * k;
        }
    }
void Arr::squarCout() {
    for(int k = --i; k >= 0; k--) {
        cout << setw(5) << array[k];
    }
}
Arr::~Arr() {
    delete [] array;
}

```

## Файл header.h

```

#ifndef Arr_h
#define Arr_h
#include <iostream>
#include <iomanip>
using namespace std;
class Arr {
    int i;
    int *array;
public:
    Arr(int x);
    void squarCout();
    ~Arr();
};
#endif

```

## Файл main.cpp

```

#include <stdlib.h>
#include <stdio.h>
#include "header.h"
#include <iostream>

```

```

using namespace std;
int main()
{
    int x;
    cin >> x;
    if(x >= 5 && x <= 20) {
        Arr* q = new Arr(x); //ВЫВОД?
        q->squarCout(); // вот тут вывод
        delete q;
    }
}

```

## Тестирование

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
11	100 81 64 49 36 25 16 9 4 1 0	100 81 64 49 36 25 16 9 4 1 0
1		100 81 64 49 36 25 16 9 4 1 0
6	25 16 9 4 1 0	25 16 9 4 1 0
2		100 81 64 49 36 25 16 9 4 1 0

