

Last updated: 2019-03-09  
vadimov@i.ua

## Практическое занятие 5 (семестр2).

Задача 5.1. Используя следующее объявление класса, создайте массив из 10 элементов и инициализируйте переменную `ch` значениями от А до J. Покажите, что массив на самом деле содержит эти значения.

```
#include <iostream>
using namespace std;
class letters {
    char ch;
public:
    letters(char c) { ch = c; }
    char get_ch() { return ch; }
};
```

Задача 5.2. Используя следующее объявление класса, создайте массив из 10 элементов, инициализируйте переменную `num` значениями от 1 до 10, а переменную `sqr` — квадратом `num`.

```
#include <iostream >
using namespace std;
class squares {
    int num, sqr;
public:
    squares(int a, int b) { num = a; sqr = b; }
    void show() { cout << num << ' ' << sqr << "\n"; }
};
```

Задача 5.3. Перепишите Example 5.5 (см. Unit 5 "Using Pointers To Objects") так, чтобы на экран выводилось содержимое массива `obj` в обратном порядке.

Задача 5.4. Измените пример Example 5.3 (см. Unit 5 "Arrays of objects") так, чтобы получить доступ к двумерному массиву через указатель. Подсказка: в C++, как и в C, все массивы хранятся непрерывно, слева направо, от младшего элемента к старшему.

Задача 5.5. Напишите программу, в которой оператор `new` используется для динамического размещения переменных типа **float**, **long** и **char**. Задайте этим динамическим переменным значения и выведите эти значения на экран. В завершение с помощью оператора **delete** освободите всю динамически выделенную область памяти.

Задача 5.6. Создайте класс для хранения своего имени и номера телефона. Используя оператор `new`, динамически выделите память для объекта этого класса и введите имя и телефон в соответствующие поля внутри объекта.

Задача 5.7. В новую версию программы Example 5.11 (см. Unit 5 "More About new And delete") добавьте деструктор и удостоверьтесь (продемонстрируйте в программе), что при освобождении памяти, обозначенной указателем ptr, для каждого элемента массива (по разу на каждый элемент массива) вызывается деструктор.

Задача 5.8. Переделайте данную программу так, чтобы в ней использовался оператор **new**:

```
char *p;  
p = (char *)malloc(100);  
// ...  
strcpy(p, " This is a test.");
```

Подсказка: Строка — это просто массив символов.

Задача 5.9. Используя оператор **new**, покажите, как динамически разместить переменную типа **double** и передать ей начальное значение -123.0987.

Задача 5.10. Создайте односвязный список (см. Unit 1 "A singly-linked list"), используя **new**, и реализуйте возможные операции с этим списком, а именно: добавление элемента, удаление элемента, копирование элемента, слияние, разбиение, вставку перед конкретным элементом или после него.

Задача 5.11. Напишите функцию **neg()**, которая меняет знак своего целого параметра на противоположный. Напишите функцию двумя способами: первый — используя параметр-указатель и второй — параметр-ссылку. Составьте короткую программу для демонстрации обеих функций.

Задача 5.12. Напишите программу, которая создает безопасный (bounded array/safe array) двумерный (2x3) массив целых. Покажите, как она работает.

Задача 5.13. Попытайтесь найти полезное применение для независимой ссылки. Напишите демонстрационную программу.

Задача 5.14. В Unit2 была создана программа с классом **strtype**, в которой память для строки выделялась динамически. Переделайте эту программу так, чтобы в ней использовались операторы **new** и **delete**.