

Задание

Это задание взято из книги «Страуструп, *Программирование: принципы и практика использования C++*» и состоит из двух частей. Первая часть даёт представление о динамических массивах и их отличии от класса `vector`.

1. Разместите в свободной памяти массив, состоящий из десяти чисел типа `int`, используя оператор `new`.
2. Выведите в поток `cout` содержимое созданного массива.
3. Освободите память, занятую этим массивом (используя оператор `delete[]`).
4. Напишите функцию `print_array10 (ostream& os, int* a)`, выводящую в поток `os` значения из массива `a` (содержащего десять элементов).
5. Разместите в свободной памяти массив, состоящий из десяти чисел типа `int`; инициализируйте его значениями 100, 101, 102 и т.д.; выведите эти значения на печать.
6. Разместите в свободной памяти массив, состоящий из одиннадцати чисел типа `int`; инициализируйте его значениями 100, 101, 102 и т.д.; выведите эти значения на печать.
7. Напишите функцию `print_array (ostream& os, int* a, int n)`, выводящую в поток `os` значения массива `a` (содержащего `n` элементов).
8. Разместите в свободной памяти массив, состоящий из двадцати чисел типа `int`; инициализируйте его значениями 100, 101, 102 и т.д.; выведите эти значения на печать.
9. Вы не забыли удалить массивы? (Если забыли, сделайте это сейчас.)
10. Выполните задания 5, 6 и 8, используя класс `vector`, а не массив, и функцию `print_vector()` вместо функции `print_array()`.

Вторая часть задания посвящена указателям и их связи с массивами. Используйте функцию `print_array()` из первой части.

1. Разместите в свободной памяти переменную типа `int`, инициализируйте её числом 7 и присвойте её адрес указателю `p1`.
2. Выведите на печать значения указателя `p1` и переменной типа `int`, на которую он ссылается.
3. Разместите в свободной памяти массив, состоящий из семи чисел типа `int`; инициализируйте его числами 1, 2, 4, 8 и т.д.; присвойте адрес массива указателю `p2`.
4. Выведите на печать значение указателя `p2` и массив, на который он ссылается.
5. Объявите указатель типа `int*` с именем `p3` и инициализируйте его значением указателя `p2`.
6. Присвойте указатель `p1` указателю `p2`.
7. Присвойте указатель `p3` указателю `p2`.
8. Выведите на печать значения указателей `p1` и `p2`, а также то, на что они ссылаются.
9. Освободите всю память, которую использовали.
10. Разместите в свободной памяти массив, состоящий из десяти чисел типа `int`; инициализируйте их числами 1, 2, 4, 8 и т.д.; присвойте его адрес указателю `p1`.
11. Разместите в свободной памяти массив, состоящий из десяти чисел типа `int`, присвойте его адрес указателю `p2`.
12. Скопируйте значения из массива, на который ссылается указатель `p1`, в массив, на который ссылается указатель `p2`.
13. Повторите задания 10–12, используя класс `vector`, а не массив.

Упражнения

1. Напишите функцию `void to_lower (char* s)`, заменяющую все прописные символы в строке `s` в стиле языка C на их строчные эквиваленты. Например, строка "Hello, World!" примет вид "hello, world!". Не используйте стандартные библиотечные функции. Строка в стиле языка C представляет собой массив символов, который завершается нулём, поэтому если вы обнаружите символ с ASCII-кодом 0 (символ `'\0'`), то это значит, что вы находитесь в конце массива.
2. Напишите функцию `char* strdup (const char*)`, копирующую строку в стиле языка C в свободную память одновременно с её выделением. Не используйте стандартные библиотечные функции.
3. Напишите функцию `char* findx (const char* s, const char* x)`, находящую первое вхождение строки `x` в строку `s`.
- ... Страуструп, *Программирование: принципы и практика использования C++*, глава 17 • Векторы и свободная память. Ответьте на **контрольные вопросы (1–12)** и постарайтесь проделать **упражнения (1–10)**.