

Раздел «Язык Си» . CoffeeArray :

- **Массивы**
 - Одномерные массивы
 - Перебор всех элементов массива
 - Явная инициализация элементов массива
 - Явная инициализация элементов массива символов
 - Многомерные массивы
 - Перебор всех элементов массива
 - Явная инициализация элементов массива
 - Типичные ошибки
 - Выход за границы массива
 - Объявление массива неизвестного размера
 - Задачи
 - Дважды напечатать входную последовательность чисел.
 - Напечатайте из заданной последовательности чисел сначала четные числа, а затем нечетные.
 - Напечатайте последовательность чисел в обратном порядке
 - Найдите индекс числа в неупорядоченной числовой последовательности.
 - Напечатайте входную последовательность, сдвинув ее циклически на 1 вправо.
 - Напечатайте входную последовательность, сдвинув ее циклически на k вправо.
 - Треугольник Паскаля
 - Решето Эратосфена

Поиск

Поиск

Раздел «Язык Си»

Главная
Зачем учить C?
Определения

Инструменты:

Поиск
Изменения
Index
Статистика

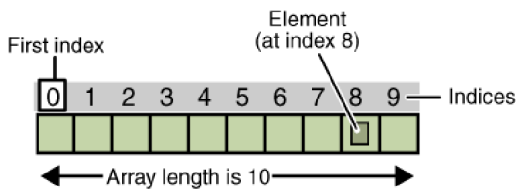
Разделы

Информация
Алгоритмы
Язык Си
Язык Ruby
Язык Ассемблера
EJ Judge
Парадигмы
Образование
Сети
Objective C

Logon>>

Массивы

- Массив – набор однородных элементов.
- Нумерация элементов начинается с 0.
- Размер массива должен быть известен на этапе компиляции. (Иначе см. Массивы переменной длины TODO)



Одномерные массивы

```
int a[10];           // объявлен массив из 10 элементов типа int.
a[2] = 7;           // записать число 7 в элемент a[2]
a[0] = a[2]-3;       // записать в элемент a[0] значение, на 3 меньше, чем хранится в (прочитано из) a[2]
```

Перебор всех элементов массива

```
int a[10];           // объявлен массив из 10 элементов типа int.
int i;               // номер очередного элемента в массиве

for (i=0; i<10; i++) {
    printf("a[%d] is %d\n", i, a[i]);
}
```

Явная инициализация элементов массива

```
int a [ ] = { 23, 7, 144 };
int b [3] = { 23, 7, 144 }; // то же самое
int c [4] = { 23, 7, 144 }; // можно, последний элемент инициализирован неявно
int d [2] = { 23, 7, 144 }; // ошибка компиляции
```

Явная инициализация элементов массива символов

- Не указывайте длину массива явно, компилятор сделает это лучше.
- Строка в Си оканчивается символом '\0'
- Можно инициализировать массив символов не набором символьных констант, а одной строковой.

```
char str1 [ ] = {'H', 'e', 'l', 'l', 'o', '\0'};
char str2 [ ] = "Hello"; // точно такой же массив из 6 элементов
```

Многомерные массивы

Перебор всех элементов массива

```
int a[3][4];           // объявлен массив из 10 элементов типа int.
int i, j;              // номер очередного элемента в массиве

for (i=0; i<3; i++) {
    for (j=0; j<4; j++) {
        printf("a[%d][%d] is %d\n", i, j, a[i][j]);
    }
}
```

Явная инициализация элементов массива

```
int a [3][4] = {
    {13,   -8,    4,   12} ,
    {-7,   -1,   14,    3} ,
    { 1,   -4,    2,   11}
};
```

Типичные ошибки

Выход за границы массива

```
int a[3];              // объявлен массив из 3 элементов типа int.
a[3] = 7;              // последний элемент - a[2]
a[-1] = 12;            // первый элемент массива - a[0]
```

Что делать? Если массив размера N, используйте элементы от a[0] до a[N-1] включительно.

Объявление массива неизвестного размера

```
int n, a[n];           // объявлен массив неизвестной длины.
```

или

```
int n;
scanf("%d", &n);
int a[n];              // объявлен массив; длина массива не известна компилятору.
```

Что делать? Внимательно читаем ограничения на входные данные.

Входные данные:

Целое число $0 < N < 1000$. Затем N чисел, по модулю не превышающих 30000, через пробел.

Значит, числа не уместятся в char, но уже поместятся в int. Этим чисел не более 1000 штук.

Значит, заводим массив из 1000 элементов типа int.

Задачи

Дважды напечатать входную последовательность чисел.

Условие задачи: Напечатайте дважды введенную последовательность чисел

Входные данные: Целое число $0 < N < 1000$. Затем N чисел, по модулю не превышающих 30000, через пробел.

Выходные данные: Входная последовательность. Еще раз входная последовательность.

Вход	Выход
3 2 17 5	2 17 5 2 17 5

```
#include <stdio.h>
#define N 1000
int main() {
    int a[N];          // сюда считываем и отсюда печатаем числа
    int i;              // номер очередного элемента в массиве
    int n;              // сколько чисел ожидается во входной последовательности

    // считываем количество чисел в n и далее сохраняем эти n чисел в массиве a
    scanf("%d", &n);
    for (i=0; i<n; i++) {
        scanf("%d", &a[i]);
    }
```

```

    }

    // печатаем n первых элементов из массива a и переводим строку
    for (i=0; i<n; i++) {
        printf("%d ", a[i]);
    }
    printf("\n");

    // печатаем n первых элементов из массива a еще раз
    for (i=0; i<n; i++) {
        printf("%d ", a[i]);
    }
    printf("\n");
    return 0;
}

```

Напечатайте из заданной последовательности чисел сначала четные числа, а затем нечетные.

Делаем как в предыдущей задаче, только в первый раз печатаем не всю последовательность, а только четные числа. Во второй раз – печатаем только нечетные.

Напечатайте последовательность чисел в обратном порядке

Никого ни с кем менять не нужно. Нужно печатать.

Допустим, мы считали 3 числа в массив a . Они хранятся в элементах $a[0]$, $a[1]$ и $a[2]$. Для печати этой последовательности в обратном порядке, подумайте:

- Сколько всего надо напечатать чисел?
- Какой элемент печатать первым?
- Какой элемент печатать последним?
- Что должно происходить с номером очередного печатаемого элемента?

Найдите индекс числа в неупорядоченной числовой последовательности.

Научимся планировать этапы решения задачи. **Этап 1.** Печатаем только индексы ВСЕХ вхождений найденного числа.

Ввод	Вывод
4 1 2 2 3 2	1 2

Этап 2. Печатаем только первый индекс найденного числа.

Ввод	Вывод
4 1 2 2 3 2	1

Этап 3. Печатаем -1, если числа в последовательности нет. Проверяем, что не поломали результаты предыдущего этапа.

Ввод	Вывод
4 1 2 2 3 77	-1
4 1 2 2 3 2	1

Напечатайте входную последовательность, сдвинув ее циклически на 1 вправо.

- Заметьте, требуют только напечатать. Ничего в считанном массиве менять не нужно.
- Сколько всего надо напечатать чисел?
- Какой элемент печатать первым?
- Сколько всего осталось напечатать чисел?
- Какой элемент печатать следующим?
- Какой элемент печатать последним?
- Что должно происходить с номером очередного печатаемого элемента?

Напечатайте входную последовательность, сдвинув ее циклически на k вправо.


Аналогично.

Треугольник Паскаля

Вы вообще этот сайт читаете? Или учебник Ворожцова?

Решето Эратосфена

Attachment	Action	Size	Date	Who	Comment
------------	--------	------	------	-----	---------

 array.png	manage	13.2 K	20 Oct 2013 - 17:03	TatyanaDerbysheva	массивы
--	------------------------	--------	---------------------	-----------------------------------	-------------------------

(с) Материалы раздела "Язык Си" публикуются под лицензией GNU Free Documentation License.