

Lecture IV:

Arrays (+ permutations)

Mark Huiskes

LIACS, Leiden University

Introduction to Programming, Media Technology MSc



Lecture Schedule

15 Sep	Lecture I	Variables, data types, operators + assignments
Today	Lecture II	Functions + mouse interaction
29 Sep		no class
6 Oct	Lecture III	Conditions + loops
Today	Lecture IV	Arrays
20 Oct	Lecture V	Classes + designing complex programs
27 Oct	Lecture VI	Recursion, data structures + sorting
3 Nov		no class
10 Nov	Lecture VII	Images and libraries
17 Nov	Lecture VIII	Max/MSP/Jitter (Edwin van der Heide)

Today

1. Tips + tricks
2. Summary sketch
3. Assignments last week
4. Array Basics
5. Permutations

Tips & Tricks

Great Programming Books

- The **C Programming Language** (2nd ed) – Kernighan and Ritchie (K&R)
- **C++ Primer** (4th ed) – Lippman
- Programming Perl (3rd ed) – Larry Wall ea
(and **many other O'Reilly books**, e.g. **Learning Perl/C#/Python**)
- The Art of Computer Programming – Donald Knuth
- Structure and Interpretation of Computer Programs – Abelson ea
<http://mitpress.mit.edu/sicp/>
- Introduction to Algorithms (2nd ed) – Cormen ea
- Beautiful Code: Leading Programmers Explain How They Think – Andy Oram
- Programming Pearls – Jon Bentley

Summary Sketch

Assignments last week

Arrays

Arrays

- An array is a data type for a group of data elements sharing a single name
- Individual array elements are accessed by indexing, e.g.

`a[0], a[2]`

- Each element has the same datatype, e.g. integer, float or boolean
- In memory an array object occupies a contiguous area

Example IV.1

Accessing array elements

- Indices are integers
- Indices should be between 0 and the length of the array – 1
- Array elements behave as regular variables

```
int[] a = {23, 4, 19, 68};
```

```
print (a[0]);  
print (" ");  
println (a[1]);
```

```
a[3] = 69;  
println(a[3]);
```

Declaring and allocating array variables

Method 1:

type[] variable-name = { elements };

Example:

```
int[] a = {8, 3, 6, 1};
```

Method 2 (the usual one):

type[] variable-name = new type[number-of-elements];

```
int[] a = new int[4];
```

.length

- In Processing (and Java) arrays are objects (next lecture)
- This has some advantages: one is the availability of a "length property"

Arrays and for-loops

(example IV.1 continued)

```
for (int i=0; i < a.length; i++)  
    print (a[i] + " ");  
println();
```

```
for (int i = a.length-1; i >= 0; i--)  
    print (a[i] + " ");  
println();
```

Example IV.2

Permutations

Permutations

- **Permutation**: a sequence containing every element of a finite set exactly once
"A permutation of n objects is an arrangement of n distinct objects in a row."
- **Example**: the permutations of the set {a, b, c}
 - abc
 - acb
 - bac
 - bca
 - cab
 - cba
- Permutations are important for analyzing sorting methods (and for shuffling cards)
- Total number of permutations of n elements: $n!$

Example IV.3

Hint for assignment IV.1

Generating permutations

mouse example