

Programming Paradigms Exercise 1 - Ruby

Johann Gamper Marco Montali Thomas Tschager

2nd Semester 2017/18

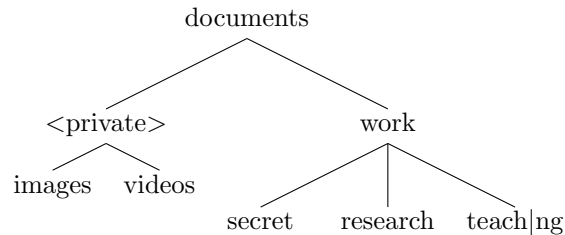
1. Implement the following methods:
 - (a) A method `arraycopyodd` that gets an array as a parameter and returns the elements found at odd positions in a new, smaller array. For the example, for the array `[5,4,7,8,3]` print 4 and 8.
 - (b) A method `arrayappend` that gets two arrays as a parameter, appends the second array to the first one and returns all the elements in a new array. For example, given the arrays `[5,4]` and `[7,8,3]`, the array `[5,4,7,8,3]` will be returned.
 - (c) A method `merge` that gets two ordered arrays as a parameter and returns a new array that is ordered and contains all the elements of the two input arrays. For example, given the arrays `[2,4,5,6,9]` and `[2,5,6,6,8]`, the array `[2,2,4,5,5,6,6,6,8,9]` will be returned.
 - (d) A method `mergesort` that gets an array as an argument and returns the array sorted in increasing order. Use the *merge sort* algorithm (see http://en.wikipedia.org/wiki/Merge_sort) and the method `merge` of the previous exercise.
2. Write a program that opens a file and outputs the file line by line with the line number in front of the line, e.g.

```
1: x = 0
2: while x < y
3:   z = compute(x)
4:   ...
```

Hints: `File.new(filename, mode)` creates a `File` object and opens a file named `filename` according to the specified mode. For example, `f = File.new("testfile", "r")` opens a file named `testfile` in read-only mode. The method `gets` fetches a line from the file (returns false if there are no more lines), while the method `close` closes the file.

Some exercises were designed by Theodoros Chondrogiannis and Michael Shekelyan

3. Assume temperature data stored in an array `t`. Write a Ruby function `maxperiod(t,x)` that calculates the length of the longest (hot) period with temperature values greater than `x`. The result is printed to the console. For example, for `t = [20, 25, 26, 23, 27]` and `x = 24`, the function prints `The longest period greater than 24 is of length 2`.
4. Write a Ruby function `palindrome(a)` to check whether an array of characters represents a palindrome, i.e., is identical to the reversed array. For instance, the array `['A', 'B', 'B', 'A']` represents a palindrom, while `['A', 'B', 'B', 'A', 'C']` does not. You are not allowed to use Ruby's array method `reverse`.
5. In this exercise we will create a class representing directories and extend the functionality of the class by including mixins of modules.
 - (a) Create a class `Directory` with the attributes `name` (a string) and `subdirectories` (a list of `Directory` objects). Ensure that objects are printed using their name (*hint*: `puts obj` uses `obj.to_s()` to convert the object to a string).
 - (b) The name of a file or a directory must not include any of the following characters: `/ > < | : &`. Create a Ruby module `FilesystemObject` with the function `isValid?` to check the whether a given directory name is valid or not. Include the module `FilesystemObject` to the mixins of the `Directory` class.
 - (c) Create a module `TreeNode` with a method `bfsprint()` which prints out all non-root tree nodes using recursive breadth-first-search and a method `getChildren` which returns the node's child nodes. Include the module to the mixins of the `Directory` class. Then add a method to the module `TreeNode` called `bfs` such that `documents.bfs(){|x| puts x}` produces the same output as `documents.bfsprint()`.
 - (d) Create an object `documents` for the following directory tree:



and output all subdirectories which do not have a valid name.