



UNIVERSITEIT VAN PRETORIA  
UNIVERSITY OF PRETORIA  
YUNIBESITHI YA PRETORIA

## DEPARTMENT OF COMPUTER SCIENCE

COS212: PRACTICAL 5

RELEASE: TUESDAY 21 MARCH 2017  
DEADLINE: FRIDAY 24 MARCH 2017, 18:00

# Instructions

Complete the tasks below. Certain classes have been provided for you in the task sub-folders of the practical download. You have been given main files which will test some code functionality, but it is by no means intended to provide extensive test coverage. You are encouraged to edit this file and test your code more thoroughly. Remember to test “corner” cases. Upload **only** the given source files with your changes in a zip archive before the deadline. Please comment your name **and** student number in at the top of each file.

## Treaps

For this practice you will need to implement the operations required for a treap to function. Please consult Section 6.10 for a description of these operations. These operations should be implemented in a class called **TreapOperations**. You have to create the relevant java file and include it in your uploads.

### Task 1: Search [10]

You have to implement the search operation. This operation has to use the given key to find and return the relevant node. If the node can not be found, null should be returned. You may assume that the provided treap will indeed be a treap.

### Task 2: Rotations [10]

You have to implement the tree rotation operations. These rotations will be used to help perform the rest of the tasks.

The first two rotations to implement the left and right rotations. These are the standard tree rotations, so nothing fancy has to be accomplished.

The next rotations is rotate up and rotate down. These rotations are specific to treaps. So the rotate up should be able to move a newly inserted leaf node up in the treap to a position dictated by its priority. Rotate down should be able to move a node down to a position where it can be removed. Note that each of these operations should only perform a single rotation at a time. The code for repeated rotations will be implemented in the next tasks.

### Task 3: insertion [10]

You have to implement the method responsible for adding new nodes. With the given main, your code should construct a treap with the same structure as the one that was manually constructed in task 1.

## Task 4: Removal [10]

You have to implement the remove method, which removes the given node from the treap. You may assume that the given node will be part of a treap.

## Submission

Submit your source files on the CS Website. Place all the files in a zip archive named as uXXXXXXXXX.zip where XXXXXXXXX is your student number. Upload your archive to the *Prac5* slots on the CS website. Submit your work before the deadline. No late submissions will be accepted.