

HashPipe

By Troels Madsen (trma)

Design choice

I choose to implement a inner class `Pipe` with a array in the same length as the height of the pipe. The array points to the other Pipes in the data.

The root pipe (boundary pipe) don't have any key or value (both set to `null`), and the initiate height of 1. The root pipe is resized if there are a new pipe with a higher height.

The methods `get()` and `floor()` contains recursive call to themselves, and uses the comparison of the keys and acts according to the return value of `.compareTo()` method. It either goes further to the next pipe and compares again, or go down in the same pipe.

Tests

It was tested with my own main method first, and after that, tested on CodeJudge to ensure correctness. I'm confident that the code works and follow both the code and performance requirement of the assignment.

CodeJudge

The code passed CodeJudge after 9 attempts. It's mainly hard to ensure that all array pointer is working and don't points out of bounds.

But the second hard thing is to keep under the performance limit. I was forced to rewrite my `put()` method completely to pass the performance tests.