# Java PriorityQueue Class

Anders Jacobsen, Dima Karaush

May 5, 2021

**Abstract**

Java's PriorityQueue class have a bottleneck when you need to update values in the queue. Depending on the use and implementation, Java's PriorityQueue can add serious performance issues when accessing the data using the poll() method. When updating a value in the queue, performance can be improved by more than 50% compared to Java's PriorityQueue implementation. Implementing your own version might remove this bottleneck from your software.

## 1 Introduction

Why did we choose this subject This article will discuss Java's "Build-in" class `PriorityQueue`. The interest for this topic has grown from an implmentation of a weighted graph in Java, which we found had a possible shortcomming for our use-case. The shortcomming was that to update a value in the queue, we had to use a linear search (loop) through the queue to update a value within. That is fine on a small scale, but what if need to draw a graph of all the cities in the world? We believe that this has space for optimization and that is why this article explores this subject.

We are going to use a previously developed `Timer` class.

How will we work with the subject

## 2 Scope

What will be in this article What will not be included in this article

## 3 Problem

The questions we will try to solve Problemfomulering

## 4 Analysis

Something about how Java PriorityQueue works now That it's missing an "update-method".

How can you make and update method How much would you gain (compared to linear update on the fly)?

## 5 Conclusion

Answer our questions and possibly introduction