

Assignment 6: The Philosophers' Waiter

Special Topics in Computer Science: Big Data Systems

CS-UH 3260 Spring 2023

MAX 10 points

In this assignment, you will implement the “Waiter” solution to the classical dining philosophers' problem using Apache Zookeeper.

1. System installation (not graded):

- Install Zookeeper on your machine, and make sure you can start the zkClient (**zkCli**)
- Install/Download the Zookeeper library in your programming of choice
 - E.g., For Python → [Kazoo](#)
- If you need help with the installation, seek help as soon as possible.


2. Description

The dining philosophers' problem is a classic computer science problem that captures the challenges of coordination between processes. The problem can be described as follows:

- Five philosophers sit at a round table, each with a bowl of noodles in front of them, and one chopstick between each pair of adjacent philosophers.
- The philosophers spend their time thinking and eating, but they can only eat if they have the two chopsticks (their right and left sides)
- However, there are only five forks available overall, and they must be shared among the philosophers.
- The challenge is to design a solution that allows the philosophers to eat without getting into a deadlock or a situation where they are all waiting for a fork that is held by another philosopher.

Many solutions have been proposed over the years, each with its own strengths and weaknesses. The **waiter** is a simple solution that relies on an external arbitrator (the waiter). In short, when a philosopher wants to eat they will ask the waiter, if the two right and left chopsticks are available, the waiter will allow them to eat, or else the waiter will reject the request.

Task: Design an algorithm that uses Zookeeper coordination service to tackle the problem.

 **Deliverable and grading:** Pseudocode with documentation explaining your solution.

- Clear code (2 points)
- Clear documentation of each step (2 points)
- Proper use of Zookeeper's API (3 points)
- A viable solution that doesn't lead to deadlocks (3 points)

 **Bonus points:**

- Implement the solution using Python (or any other language) using Zookeeper and its library. (2 points)