

Operating Systems (coe628)

Lab 9—Dining Philosophers

April 3, 2017

Duration 2 weeks

Objectives

- Deadlock and the dining philosophers
- Solving deadlock with semaphores (standard Java)

Notes

- The dining philosopher problem is described [here](#). (These are lecture notes from the University of Southern California, UCB.) This paper also includes a C version of the algorithm using semaphores.
- The [Wikipedia article](#) also describes the problem and gives a solution in python.

Part A: Translate the C version to Java

- Convert the UCB C algorithm to Java.

Part B: Implement your own version of the Semaphore class

- Use your version of Semaphore from the previous lab.

And Finally: Submit your lab

To submit your lab do:

1. Zip your source code files into a file named `lab9.zip`.
2. Submit the zip file with the command: `submit coe628 lab9 lab9.zip`

Copyright © 2016 Ken Clowes. This work is licensed under the Creative Commons Attribution 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.