CMSI 387-01

Operating Systems

Spring 2013

Assignment 0411 Feedback

Allyson Pascua

All of Time and Space

(updated feedback reflects commits up to April 23)

3b — You did a good job with tracking down your selected papers and recording their connections and differences. The Concurrent Pascal paper is a particularly seminal find, combining the work of some really big guns (Hansen, Wirth, etc.). Your writeups could have had more detail though, and you did not address the issue of the credentials of each author. Had you done that, Per Brinch Hansen would have stood out significantly among the others. (|)

4d — Excellent work with finding papers from available sources. (+)

4e — Your commits and messages are appropriate for the work performed. (+)

4f — Not submitted on time. (–)

The Dining Philosophers Problem

(updated feedback reflects commits up to April 21)

2d — You have completed a beautifully rendered implementation of the dining philosophers problem. Your approach definitely fulfills the "state display" request in the assignment, but internally it is missing one critical element: *actual concurrency*. This is partially due to your choice to implement this as a web app, but the change in language in itself is not the issue. The true issue is that *you don't actually have threads running*. Instead, you call `setInterval` on a function that *sequentially* cycles through each philosopher's state and actions. Your implementation is effectively a single-threaded loop, and as a result, you don't need any synchronization primitives, plus there is actually no applicable context within which to include error-checking code, which is the other specific request given in the assignment (i.e., there is no code to add/remove that will show how synchronization primitives are indeed addressing the critical section). Unfortunately, this makes you miss a significant portion of the intended knowledge and skill set that is supposed to be acquired from this assignment. (/)

4a — Your code is certainly functional and works as intended. You do miss some of the specific best practices that we have established for JavaScript, such as correct semicolon placement, use of === instead of ==, and encapsulation within an anonymous function. Based on those, and not the non-threaded implementation, the proficiency takes a hit. (|)

4b — Separation of concerns is generally good except for the "bare" variables defined in top-level namespace. Wrapping in an anonymous function would solve this. (|)

4c — Your code is overall highly readable and clear, with just a few spacing or indentation inconsistencies. Inline comments are provided in appropriate places. (+)

4d — You did a good general job in figuring out how to put this web app together, even if, as mentioned, it misses an important point of the exercise. However, that has already been accounted for in *2d*, and so this outcome can go fairly unscathed. (+)

4e — Good version control habits are well on display here. (+)

4f — Not submitted on time. (–)