CMSI 387-01

OPERATING SYSTEMS

Spring 2014

Assignment 0422 Feedback

Michael A. Fraser *mfraser42 / mfraser702@gmail.com*

Dining Philosophers

2d — Nihilist philosophers. (–)

4a — Non-existential code. (–)

4b — Code has no form. (–)

4c — No "codito" ergo no sum. (–)

4d — Based on this, I think it's fair to say that you haven't been able to sufficiently assimilate the process synchronization material. In other words, theory is one thing but practice is another **:)** (–)

4e — Err…no "commito" ergo no sum? (–)

4f — Not submitted on time (sorry, an initial copy doesn't count). (–)

Paged Memory Address Translation

2d — You have successfully implemented paged memory address translation from the ground up. (+)

4a — No issues with your code outside of that unnecessary bitwise-`&`. Otherwise works as spec'ed. (+)

4b — One little separation of concerns tweak would be unhardcoding 256. But overall n.b.d. (+)

4c — Clean code, no big issues. Your spacing isn't 100% consistent, and that stands out more due to the shortness of the code. Just letting you know that I still noticed **O_o** (+)

4d — You successfully utilized available information to implement paged memory address translation. (+)

4e — Commit frequency and messages are appropriate for the work done. (+)

4f — Submitted on time. (+)

Updated feedback for Dining Philosophers based on commits up to 5/9/2014; only re-evaluated outcomes are included:

2d — The philosophers thank you for feeding them **:)** (+)

4a — You've gotten the core solution right, with decent output for displaying the system state (though still not perfect—see inline comments). Unfortunately, you still don't include any "sanity check" code or asserts that cry foul upon improper behavior. Kind of a major thing that still keeps this from getting any higher, unfortunately. Oh, and `atoi` **:)** (/)

4b — Code is decently structured now with logically separated/sequenced sections, discernible even in the single-file versions. Multifile version would be even better separated, yes, but requires more work. (+)

4c — Code is decently formatted with good consistency. (+)

4d — You did pretty well finding needed information but still missed the error checking, which is weighted quite heavily. (|)

4e — Commit frequency and messages are appropriate for the work done. (+)