CMSI 387-01

OPERATING SYSTEMS

Spring 2014

Assignment 0422 Feedback

Carlos Agudo *r2d32 / carlos.jorrillo@gmail.com*

Dining Philosophers

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

4a — You've gotten the core solution right—but you need to make it clear to the user that this is the case. Thus, suggestions given inline are primarily refinements to output format so that the user can perceive what's happening more easily, particularly with respect to errors. Speaking of errors, you include "sanity check" code when picking up a fork, but not when putting it down. Plus, you should *exit* the program upon detecting the violation, because the point of this exercise is to guarantee that it *never* happens. More details can be found in the inline comments. (|)

4b — Code is decently structured with logically separated/sequenced sections, despite the overall monolithic single-file structure of the program. (+)

4c — I was able to read your code pretty easily, though you do have the occasional spacing inconsistency. Fortunately not enough to mess things up. (+)

4d — Your work demonstrates good use of the information available to you, except for the full understanding of the role played by good program output plus a complete awareness of which parts of the program need a sanity-check block. (|)

4e — Just two commits for this fairly involved, multithreaded program? Not so sure about that. At least your messages are decent. However, you also committed *a.out*. No no no. (/)

4f — Initial [very thin] skeleton submitted 6 days after the due date, with work up to May 1. (/)

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 — Two separation of concerns issues: first, including a *.c* file in another is *not* the way to manage multiple-file C programs. Did you see that in *any* of the sample code shown, particularly the bounded buffer example? Even though it compiles, that doesn't mean you should do it. Did you see any *.c* includes in the bounded buffer sample??? That should have given you a hint. Plus *Makefile* too. Second, you should unhardcode the maximum address check. That header file is there for a reason! (/)

4c — For something this brief, even small spacing/indentation hiccups get pretty magnified. Most of your code is cleanly formatted, but the spacing goes crazy in line 36 of *addressTranslation.c*. What happened there? You should have noticed that this looked significantly different from the rest of the code. (|)

4d — You successfully utilized available information to implement paged memory address translation, with issues being more a matter of general code practices rather than the algorithm itself. (+)

4e — Commit frequency and messages are appropriate for the work done, but build product (*a.out*) was committed. Don't. Commit. Build. Products! (/)

4f — Completed 2 days after the due date (initial commit was just a copy, doesn't count). (|)