CMSI 387-01
OPERATING SYSTEMS
Spring 2014

Assignment 0422 Feedback

Marc A. Mouallem *MarcMouallem / me@marcmouallem.com*

Dining Philosophers

2d — The philosophers thank you for feeding them well—or are you??? (see the long inline comment in *Philosopher.cpp*) (**|**)

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. And speaking of errors, you don't include any "sanity check" code or asserts that enforces proper behavior. This is the more significant issue, especially because it was specifically requested by the assignment and yet there is no sign of it here. More inline comments give hints on how/where to do this. Finally, why use `sem_t` rather than `pthread_mutex`? That's overkill. (**/**)

4b — Code is decently structured with logically separated/sequenced sections. Using C++ does no harm but doesn't help either—there's nothing in this implementation that makes a case for C++ being more compelling than standard C. (**+**)

4c — I was able to read your code pretty easily. (**+**)

4d — Your work demonstrates good use of the information available to you, except for the full understanding of how important the error checking is and the questionable choice to use `sem_t` as your synchronization primitive rather than `pthread_mutex`. (**|**)

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

4f — Submitted around 1 year and 11 months after the original due date. (**–**)

Paged Memory Address Translation

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

4a — No incorrect issues with your code; just some that detract oh so slightly from efficiency (remember, this computation will happen with virtually every machine instruction, so for once every little bit *does* count). First is that unnecessary bitwise-`&`; second is that extraneous condition that checks for `ERR_OUT_OF_RANGE`. 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. (**+**)

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

4e — One commit—hmmmm…I *guess* this is small enough that you can do it all in one commit, but I would say the pro way is to have at least two—one for the initial copy of the code, then the second for your fix to it. (**|**)

4f — Submitted just under two years after the original due date. (**–**)