

Montserrat Alonso

Taiki Tsukahara

Hailu Xu

CECS-326 Operating Systems

15 March 2024

Synchronization

For our project we decided to do it in C++ because we had experience in using pthreads in our CECS-325 class. In our program we decided that we should make symbolic constants and macros so it can be easier to maintain. For example, in order to keep track of the left and right fork's indexes of a certain philosopher we created the macros LEFT and RIGHT. We knew that we would have to use mutex because of what was said in our requirements however we still proceeded to not use it in the beginning and that led to a deadlock. Another thing that led to a deadlock was in our while statement: `while (state[LEFT] == EATING || state[RIGHT] == EATING) pthread_cond_wait(&cond_var, &mutex)`. If we changed the while statement to an “and” statement then it would eventually lead to a deadlock and so that's why we had to change it to an “or” statement. In the main function we initialize all of the 5 philosophers to the thinking state and then when we create threads for the philosophers, we then execute the functions of `pickup_forks` and `return_forks`, where we change the state to HUNGRY and then to EATING. This then goes on forever because of our statement of `while(true)`, so each philosopher will be going through each state.