OS Assignment 5

Group Details

Group - 103 **Members** : { Arjun Tandon <2022095>, Chahat Ahuja <2022138> }
https://github.com/arjun22095/os-assignment/tree/main/group103-simple-multithreader

Implementation

1. There are two functions that need to be implemented as mentioned in PDF, one is the Simple-Parallel-For (SPF) and other is Nested-Parallel-For (NPF)
2. Regrading SPF
    1. The main handler function, first starts measuring the time using `std::chrono`
    2. It then creates a `pthread_t` array of `numThread` length
    3. Calculates the chunk size based on total loop range and `numThreads`
    4. It then creates the args struct list
    5. In a loop it uses `pthread_create` to create a thread `[j]` that is given the args cur_args `[j]` and calls the thread_func for the SPF
    6. The thread_func for SPF, it just casts the arg struct pointer from `void *` to `args *`
    7. It then checks whether the thread is the last thread or not and accordingly executes execution
3. Regrading NPF
    1. The main handler function, first starts measuring the time using `std::chrono`
    2. It then creates a `pthread_t` array of `numThread` length
    3. Calculates the chunk size based on total loop range and `numThreads`
    4. It then creates the `args2` struct list
    5. In a loop it uses `pthread_create` to create a thread `[j]` that is given the args cur_args `[j]` and calls the thread_func for the PPF
    6. The thread_func for PPF, it just casts the arg struct pointer from `void *` to `args2 *`
    7. It then calculates based on the `thread_no` and the `start` and `end` values, the range of `i` and `j` for the current thread using a simple formula. Its `args2` struct doesn't directly tell it the starting point and ending point of it's execution, it has to be calculated from the `thread_no` which is passed
    8. It then checks whether the thread is the last thread or not and accordingly executes execution