Lab 2

Konrad Rej
Dang Tuan Phong Nguyen

Data structure
In the thread data structure we added a variable called ticksToBlock which keeps track of the remaining ticks the thread is supposed to be blocked.

Algorithms
Instead of using the functionality of busy waiting, we simply change the status of the thread to block and save the necessary data (amount of ticks to be blocked) so we know how much time is left for the thread to remain blocked before returning to the ready state. Each time the timer interrupt runs, we enter a critical section, therefore we disable interrupts before it and restore them afterwards. In the critical section we iterate over each thread and if it is blocked we decrement remaining ticks and check whether to unblock the thread yet or not.

Synchronization
To prevent other threads from running during our critical section (calls to thread_block and thread_foreach) we turn off interrupts before entering it and then upon leaving it we restore the previous state of interrupts.

Rationale
We decided to save the amount of ticks left to be blocked in the data structure (thread) instead of in a global context to keep the data relevant to a specific thread contained in the same structure. The algorithm loops through n threads and checks if statements if the function should execute thread unblocking or decrement the tick counter, and the time complexity would be relatively low.