MP5 Design Document

1.Components Implemented

- Thread constructor and management
- Scheduler framework (abstract base class)
- Round Robin scheduler implementation
- Thread dispatching mechanism
- Kernel thread initialization

2. Implementation Details

- Thread Constructor and Management
    - Creates threads with a dedicated stack
    - Initializes thread context for execution
    - Sets up stack frame for proper thread startup and termination
    - Maintains thread ID management
- Scheduler Framework
    - Abstract base class defining the scheduler interface
    - Provides virtual functions for thread management:
    - yield: Allow thread to give up CPU
    - resume: Add thread to ready queue
    - add: Include new thread in scheduling
    - terminate: Handle thread completion
    - get_next_thread: Select next thread to run
- Round Robin Scheduler
    - Implements scheduler using FIFO queue
    - Equal time slices for all threads
    - Simple preemptive scheduling
- Thread Dispatching
    - Context switching between threads
    - Low-level register and stack management
    - Proper CPU state preservation
    - Support for timer-based preemption
- Kernel Thread Integration
    - Initialization of first thread
    - Thread lifecycle management
    - System bootstrap process

3. Option 1 & 2 implementation

- The switch between FIFO and RR is at the line 396 of kernel.C. The initial state is FIFO, if commented out, the scheduler will be switched to RR.