15-Day LibFT Learning and Implementation Schedule

Daily Fixed Schedule

- 07:00 - 08:00: Morning routine
- 08:00 - 09:30: Gym session
- 10:00 - 13:00: Main LibFT work block
- 13:00 - 14:00: Lunch break
- 14:00 - 16:00: Side project
- 16:30 - 19:00: Second LibFT work block
- 19:00 - 20:00: Review and planning for next day

Day-by-Day Plan

Week 1: Fundamentals and Basic Functions

Day 1: Setup and Memory Basics

- **Morning Block:**
  - Set up development environment
  - Study memory management concepts
  - Read man pages for memset, memcpy
- **Side Project Time**
- **Evening Block:**
  - Implement: ft_memset, ft_bzero
  - Create basic test cases

Day 2: More Memory Functions

- **Morning Block:**
  - Study pointer arithmetic
  - Review memcpy and memmove concepts
- **Side Project Time**
- **Evening Block:**
  - Implement: ft_memcpy, ft_memmove
  - Write test cases

Day 3: String Basics

- **Morning Block:**
  - Study string concepts
  - Review string.h functions
- **Side Project Time**
- **Evening Block:**
  - Implement: ft_strlen, ft_strlcpy, ft_strlcat
  - Test implementations

Day 4: Character Functions

- **Morning Block:**
  - Study ASCII table
  - Review character classification
- **Side Project Time**
- **Evening Block:**
  - Implement: ft_isalpha, ft_isdigit, ft_isalnum, ft_isascii, ft_isprint
  - Implement: ft_toupper, ft_tolower
  - Test all char functions

Day 5: String Search

- **Morning Block:**
  - Study string search algorithms
- **Side Project Time**
- **Evening Block:**
  - Implement: ft_strchr, ft_strrchr, ft_strncmp
  - Create comprehensive tests

Week 2: Advanced Functions

Day 6: Memory Search and Compare

- **Morning Block:**
  - Study memory comparison techniques
- **Side Project Time**
- **Evening Block:**
  - Implement: ft_memchr, ft_memcmp
  - Write test cases

Day 7: String to Number

- **Morning Block:**
  - Study atoi implementation
  - Review edge cases
- **Side Project Time**
- **Evening Block:**
  - Implement: ft_atoi
  - Create extensive test cases

Day 8: Memory Allocation

- **Morning Block:**
  - Study dynamic memory allocation
  - Review calloc implementation
- **Side Project Time**
- **Evening Block:**

    - Implement: ft_calloc, ft_strdup
    - Test memory leaks

Day 9: String Manipulation Part 1

- **Morning Block:**
    - Study advanced string operations
- **Side Project Time**
- **Evening Block:**
    - Implement: ft_substr, ft_strjoin
    - Write test cases

Day 10: String Manipulation Part 2

- **Morning Block:**
    - Review string parsing
- **Side Project Time**
- **Evening Block:**
    - Implement: ft_strtrim, ft_split
    - Create test suite

Week 3: Final Functions and Bonus

Day 11: Number to String and Function Pointers

- **Morning Block:**
    - Study number conversion
    - Review function pointers
- **Side Project Time**
- **Evening Block:**
    - Implement: ft_itoa, ft_strmapi, ft_striteri
    - Test implementations

Day 12: File Descriptors

- **Morning Block:**
    - Study file descriptors
- **Side Project Time**
- **Evening Block:**
    - Implement: ft_putchar_fd, ft_putstr_fd, ft_putendl_fd, ft_putnbr_fd
    - Test with different file descriptors

Day 13: Bonus Part - Linked Lists 1

- **Morning Block:**
    - Study linked list concepts
- **Side Project Time**

- **Evening Block:**
  - Implement: ft_lstnew, ft_lstadd_front, ft_lstadd_back, ft_lstsize, ft_lstlast
  - Create list manipulation tests

Day 14: Bonus Part - Linked Lists 2

- **Morning Block:**
  - Review list operations
- **Side Project Time**
- **Evening Block:**
  - Implement: ft_lstdelone, ft_lstclear, ft_lstiter, ft_lstmap
  - Complete bonus tests

Day 15: Final Review and Testing

- **Morning Block:**
  - Complete full test suite
  - Memory leak checks
- **Side Project Time**
- **Evening Block:**
  - Final optimizations
  - Documentation
  - Prepare for evaluation

Tips for Following This Schedule:

1. **Be Flexible:** Adjust timings based on your progress and energy levels
2. **Track Progress:** Mark completed functions and concepts
3. **Test Continuously:** Write tests as you implement each function
4. **Stay Consistent:** Maintain the gym routine for better focus
5. **Review Daily:** End each day by reviewing progress and adjusting next day's plan
6. **Side Project Balance:** Keep side project scope manageable during this period

Priority Rules:

- If running behind, focus on mandatory functions first
- If a concept is unclear, spend extra time on understanding before implementation
- If ahead of schedule, use extra time for more testing and optimization
- Never skip gym - it helps maintain mental clarity

Emergency Buffer:

If needed, you can use side project time for LibFT on days when you're behind schedule, but try to maintain balance when possible.