

F.R.I.D.A.Y || MPX Project User Manual

```

*****          *****          **          **          ****
**              **      **        **      **              **      **
**              **      **        **      **              **      **
****           **   **            **      **              **      **
**             ****              **      **              ****              **
**             **   **            **      **              **      **
**             **       **         **      **              **      **
**             **       **         **      **              **      **
**             **       **         **      **              **      **
**             **       **         **      **              **      **
Welcome to MPX. Please select an option
=> help
=> get-time-date
=> alarm HH:mm:ss
=> set-time HH:mm:ss
=> set-date MM/DD/YY
=> set-timezone
=> dragonmaze
=> load-R3
=> version
=> shutdown
=> color
=> clear
=> allocate-memory
=> free-memory
=> show-allocate
=> show-free

```

Table of Contents

1 PREFACE	3
1.1 Description of Project	3
1.2 Project Details	3
1.3 Overview of Project	3
2 MAIN COMMANDS	4
2.1 Command Handler	4
2.2 Version Command	4
2.3 Help Command	5
2.4 Shutdown Command	6
2.5 Set Date Command	6
2.6 Set Time Command	7
2.7 Set Timezone Command	7
2.8 Get Time Date Command	8
2.9 Color Command	8
2.10 Clear Command	9
2.11 Load R3 Command	9
2.12 Alarm Command	10
3 PCB COMMANDS	11
3.1 PCB Command	11
3.2 PCB Delete Command	11
3.3 PCB Block Command	11
3.4 PCB Unblock Command	11
3.5 PCB Suspend Command	12
3.6 PCB Resume Command	12
3.7 PCB Priority Command	12
3.8 PCB Show Command	12
3.9 PCB Show-Ready Command	13
3.10 PCB Show Blocked Command	13
3.11 PCB Show All Command	14
4 HEAP COMMANDS	14
4.1 Allocate Memory Command	14
4.2 Free Memory Command	15
4.3 Show Allocate Command	15
4.4 Show Free Command	16

1 PREFACE

1.1 Description of Project

Welcome to our CS 450 Group MPX project nicknamed F.R.I.D.A.Y. In this user manual you can find helpful information about each command that has been implemented as well as a look into the mind of the designers to learn more about how each command has been implemented.

1.2 Project Details

In this project, the students must follow the Modules R1 through R6 and implement the functions given in each specific module. Upon completion of each module, the students will continue to work toward a final project that can serve as a demonstration of their working knowledge of OS concepts.

1.3 Overview of Project

R1: In the early phases of this project the students are working toward creating an Interface and implementing a series of commands that can be used in the command line and executed to provide output.

R2 : For the second phase of this project the team has to start implementing PCB commands that can be called in the command line, or get explained to the user in the help menu. These commands help the user interact with the PCB linked list and change it at will.

R3 : For the third phase of the project the team started implementing changes that allows processes in the PCB queue to be dispatched and executed and allows the system to switch to different processes after the running process gives control back using context switching.

R4 : Once the team has successfully implemented the dispatching concepts created in module R3, the team shifted their focus on applying these concepts within their projects. Module R4 is used for the team to implement an alarm command that is responsible for using the dispatching logic created in R3 so the project can create a process with an attached time and message. This command utilizes the dispatching concepts practiced in R3 by dispatching the created alarm at the correct time.

R5 : For the fifth phase of the project we had to implement our own heap free and allocate functions that will be used to dynamically allocate and free memory for our operating system. This allows us to reclaim memory that was allocated.

R6 : Updated Later

2 MAIN COMMANDS

2.1 Command Handler

The command handler serves as the primary way users will interact with the operating system. The goal of the command handler is for the user to have a menu of options that they can choose to execute within the operating system. The command handler is set up so the user can set the time, set the date, print the date and time, get the version history, shutdown, change the color, and clear the MPX program.

```
Welcome to MPX. Please select an option
=> help
=> get-time-date
=> set-time HH:mm:ss
=> set-date MM/DD/YY
=> set-timezone
=> version
=> shutdown
=> color
=> clear
```

2.2 Version Command

In the Version command, the user will be able to print the current version of MPX as well as the date in which they are executing the command. The version command waits for user input, verifies the user is selecting the version command, and prints the current module of the project as well as the current date and time of compilation.

```
>> version
Module: R2
Feb 17 2023
09:46:07
```

2.3 Help Command

In the Help command, the user will be able to see basic instructions for each command as well as how to execute each of these commands. The help command waits for user input, verifies the user is selecting the help command, and will print out a list of helpful instructions that the user can use in the event of a misunderstanding.

```
If you need help, enter one of the Statments below!
=> enter 'help set-time'
=> enter 'help set-date'
=> enter 'help get-time-date'
=> enter 'help help'
=> enter 'help set-timezone'
=> enter 'help alarm'
=> enter 'help version'
=> enter 'help shutdown'
=> enter 'help color'
=> enter 'help clear'
=> enter 'help pcb'
```

All the base Help Commands

In R2, the Help pcb function was added, with its own sub menu. It works the same as the original command, but prompts a specific menu for all the pcb commands. Once they call a help command, it prints the basic functionality of the command.

```
>> help pcb
The 'pcb' command shows all the pcb commands available to the user. the help commands are listed below
=> enter 'help pcb create'
=> enter 'help pcb delete'
=> enter 'help pcb block'
=> enter 'help pcb unblock'
=> enter 'help pcb suspend'
=> enter 'help pcb resume'
=> enter 'help pcb priority'
=> enter 'help pcb show'
=> enter 'help pcb show-ready'
=> enter 'help pcb show-blocked'
=> enter 'help pcb show-all'
```

All the PCB help commands

2.4 Shutdown Command

In the shutdown command, the user will be able to exit the command handler and return to a baseline terminal. The user will be able to execute the shutdown command from anywhere within the command handler. If the user calls a different command first, they will still be able to execute the shutdown command and will exit the command handler loop from the current command they are executing. As a confirmation step, the user will have to confirm in the command that they in fact do want to shut down the project.

```
>> shutdown
Are you sure you want to shutdown? (y/N): y
klogv: Starting system shutdown procedure...
klogv: Shutdown complete.
```

Successful Shutdown

```
>> shutdown
Are you sure you want to shutdown? (y/N): N
>>
```

Shutdown Command Canceled

2.5 Set Date Command

In the set date command, the user will be able to change the date stored in the OS if it is different from the date that will be automatically generated. The user can call this command by typing set date into the command handler and using this format “Set Date MM/DD/YY” to successfully change the date stored in the OS. This command will check that the user enters the command properly, will save the date that was inputted, and will print a confirmation message upon successful entry.

```
>> Set-Date 09/26/23
Set the date to: 09/26/23
>> █
```

Changing the Date in the set-Date Command

2.6 Set Time Command

In the set time command, the user will be able to change the time stored in the OS if it is different from the date that will be automatically generated. The user can call this command by typing set time into the command handler and using this format “Set Time ##:##:##” to successfully change the time stored in the OS. This command will check that the user enters the command properly, will save the time that was inputted, and will print a confirmation message upon successful entry.

```
>> Set-Time 12:30:23
Set the time to: 12:30:23
>> █
```

Changing the time in the set-time command

2.7 Set Timezone Command

In the set timezone command, the user will be able to choose from a list of time zones and can select their current time zone to be updated in the operating system. Once the user enters “Set Timezone” a list of common time zones will be displayed in the terminal. The user will then be able to enter the abbreviation of any listed time zone and that time zone will be used as the current time saved in the OS. Upon successful selection of a new time zone, the user will be prompted with a message confirming that their time zone has been changed and will reflect their newly selected time zone.

```
>> Set-Timezone
What time zone do you want to be in?
=> ET (Eastern Time - New York)
=> CT (Central Time - Chicago)
=> MT (Mountain Time - Denver)
=> PT (Pacific Time - Los Angeles)
=> AKT (Alaskan Time - Anchorage)
=> HAT (Hawaii-Aleutian Time - Honolulu)
=> UTC (Universal Time Code - London)
=> CET (Central European Time - Paris)
=> AT (Arabian Time - Moscow)
: ET
Set the timezone to 'Eastern Time'!
>>
```

All of the Available time zones

2.8 Get Time Date Command

In the get date/time command, the user will be able to see the current date, day of the week, and time of day the command is being executed at. The get date/time command will default to the timezone of ET and will display information based on this time zone unless otherwise specified in the set time/date/timezone commands listed above. In the event that the user executes a set command prior, the get time/date command will print the information based off of the input from the commands used prior.

```
>> get-time-date
Friday, 02/17/23 @ 14:47:08 UTC
```

Get Date/Time command with standard time

```
Set the timezone to 'Arabian Time'!
>> get-time-date
Friday, 02/17/23 @ 16:47:45 AT
```

Get Date/Time command after changing system time

2.9 Color Command

In the Color Command, the user will be able to change the color of their font, and will stay that way until it is eventually changed or they shutdown the system. There are various color options given to the user, including black, bright-black, red, bright-red, green, bright-green, yellow, bright-yellow, blue, bright-blue, magenta, bright-magenta, cyan, bright cyan, white, and bright-white. To restore the color back to the default color, the user needs to reprompt the color command, and then hit 'reset'.

```
=> black
=> bright-black
=> red
=> bright-red
=> green
=> bright-green
=> yellow
=> bright-yellow
=> blue
=> bright-blue
=> magenta
=> bright-magenta
=> cyan
=> bright-cyan
=> white
=> bright-white
=> reset
```

All Available Color Options Shown

```
Set the color to 'red'!
>> set-timezone
What time zone do you want to be in?
=> ET (Eastern Time - New York)
=> CT (Central Time - Chicago)
=> MT (Mountain Time - Denver)
=> PT (Pacific Time - Los Angeles)
=> AKT (Alaskan Time - Anchorage)
=> HAT (Hawaii-Aleutian Time - Honolulu)
=> UTC (Universal Time Code - London)
=> CET (Central European Time - Paris)
=> AT (Arabian Time - Moscow)
.
```

The Color Command Working in Action

2.10 Clear Command

The Clear Command Will clear the Terminal, and give the user a fresh screen. The Clear command can be shown in the command handler, and when entered, will clear whatever prompts the user has done recently.

```
Welcome to MPX. Please select an option
=> help
=> get-time-date
=> set-time HH:mm:ss
=> set-date MM/DD/YY
=> set-timezone
=> version
=> shutdown
=> color
=> clear
>>
```

Clear Being Shown in the Command Handler

```
>>

```

What The Terminal looks like after the clear command is executed

2.11 Load R3 Command

The Load R3 command will create the processes, and load them into a queued non-suspended state. it loads all five processes, and dispatches them in ascending order one at a time. After dispatching the 1-5 processes, it will cycle back and dispatch 2-5 in order; The next cycles are 3-5, 4-5, and then the 5th processes respectively.

```
>> load-r3
Created process named 0!
Created process named 1!
Created process named 2!
Created process named 3!
Created process named 4!
proc1 dispatched
>>
```

The Load-R3 Command Creating the processes and dispatching process 1

2.12 Alarm Command

The alarm command allows the user to create alarms that will go off in the future. The code to use the alarm command is 'alarm HH:mm:ss', with the format being hours: minutes: seconds. Once the user enters a designated time to display the alarm, the user is prompted to add a message to be displayed along with the alarm. The alarm command works while the user is idle in the operating system.

```
>> alarm 13:55:00
Enter the message to be saved in the alarm (Maximum 50 chars):
> First Alarm
Set an alarm for: 13:55:00 with the message First Alarm
```

The Alarm Being Created

```
>>
proc4 dispatched
>>
First Alarm
>>
proc5 dispatched
```

The Alarm Going Off

3 PCB COMMANDS

3.1 PCB Command

The PCB command is split into many different sub-commands to allow the user to interact with the PCB queue. The main command 'pcb' does not have any function other than to notify the user that they must specify a proper sub command. All subcommands are listed in the 'help pcb' command's output.

3.2 PCB Delete Command

The Delete subcommand is used for deleting an existing PCB from the PCB queue. If the PCB you're trying to delete doesn't exist, nothing will happen. System processes cannot be deleted.

```
>> pcb delete pcbblock  
Removed PCB named 'pcbblock'!
```

Deletion of a PCB

3.3 PCB Block Command

The Block command is used for setting the state of a PCB to blocked. If the PCB is already blocked or doesn't exist, nothing will change.

```
>> pcb block list  
The pcb named: list was blocked
```

Blocking a PCB

3.4 PCB Unblock Command

The PCB unblock command will unblock a currently blocked pcb. If there is not a blocked pcb, it will just remain unblocked and nothing will change. An example of the unblock command working is shown below.

```
>> pcb unblock list  
The pcb named: list was unblocked
```

Unblocking a PCB

3.5 PCB Suspend Command

The PCB suspend command will suspend a PCB from execution entirely, marking it with the 'suspended' dispatch state. System class PCBs cannot be suspended.

```
>> pcb suspend pcblock  
The pcb named: pcblock was suspended
```

Suspension of a PCB

3.6 PCB Resume Command

The PCB resume command will resume an already suspended PCB, changing the dispatch state to 'not suspended'.

```
>> pcb resume pcblock  
The pcb named: pcblock was resumed
```

Resuming of a PCB

3.7 PCB Priority Command

The PCB priority command will change the PCB priority. This command will change priority with the most important priority being 0, and the least priority being 9. The priority is originally defined when the PCB is created, but if the user wants to change it, it needs to be defined here.

```
>> pcb priority list 6  
The pcb named: list was changed to priority 6
```

PCB changing priority

3.8 PCB Show Command

The show command will display the contents of the given PCB name such as the name, priority class, state, and whether it's suspended.

```
>> pcb show thing  
PCB "thing"  
- Priority: 1  
- Class: User  
- State: Blocked  
- Suspended: Not Suspended
```

PCB named "thing" displayed

3.9 PCB Show-Ready Command

The show-ready command will display all the PCBs that are currently in the ready state.

```
>> pcb show-ready
PCB "thing"
- Priority: 1
- Class: User
- State: Ready
- Suspended: Not Suspended
PCB "apple"
- Priority: 3
- Class: User
- State: Ready
- Suspended: Not Suspended
```

PCBs in ready state

3.10 PCB Show Blocked Command

The PCB showblocked command will show all of the PCBs that are currently blocked. An example of this is shown below.

```
The pcb named: list3 was blocked
>> pcb show-blocked
PCB "list3"
- Priority: 3
- Class: User
- State: Blocked
- Suspended: Not Suspended
PCB "list2"
- Priority: 8
- Class: System
- State: Blocked
- Suspended: Not Suspended
```

All Blocked PCB's

3.11 PCB Show All Command

The PCB Show all command will show all the PCB commands, no matter the state of the PCB, the class or if it's suspended or not. An example of this is shown below.

```
>> pcb show-all
PCB "list"
- Priority: 6
- Class: User
- State: Ready
- Suspended: Not Suspended
PCB "list3"
- Priority: 3
- Class: User
- State: Blocked
- Suspended: Not Suspended
PCB "list2"
- Priority: 8
- Class: System
- State: Blocked
- Suspended: Not Suspended
```

The PCB Show all Command Working in Action

4 HEAP COMMANDS

4.1 Allocate Memory Command

The Allocate Memory Command will allocate the specified amount given by the user. This specified value will be allocated into the next available memory block. An example of the Allocate Memory Command is Shown Below

```
>> allocate-memory
How many bytes do you want to allocate?
2
Successfully Allocated The 2 Number of Bytes at 0xD001154!
IDLE PROCESS EXECUTING.
```

Allocating 2 bits to the next block

4.2 Free Memory Command

The Free memory will free all of the memory that is given by the user. If the address of the memory given is not available, no memory will be freed. The example of the free memory command is shown below

```
>> free-memory D00118B
Successfully freed memory
IDLE PROCESS EXECUTING.
```

Freeing the bits in memory

4.3 Show Allocate Command

The Show allocate command will display all of the allocated memory blocks, its memory starting address, and its size. The example of how the show allocate command works is shown below.

```
>> show-allocate
Memory Control Block List Allocated

Memory Block #0
Memory Start: 0xD000010
Size: 20

Memory Block #1
Memory Start: 0xD000034
● Size: 2080
○

● Memory Block #2
Memory Start: 0xD0000864
○ Size: 8
○

Memory Block #3
Memory Start: 0xD000087C
Size: 8

Memory Block #4
Memory Start: 0xD0000894
Size: 2080

Memory Block #5
Memory Start: 0xD00010C4
Size: 5
```

the first 6 memory blocks, its memory start, and size

4.4 Show Free Command

The Show free command will show the memory control block that is freed, and where the memory start is in hexadecimal. The command will also show the size of the block- an example of this is shown below.

```
>> show-free  
Memory Control Block List Free  
  
Memory Block #0  
Memory Start: 0xD00114E  
Size: 45570
```

The Memory block that's freed. Shows the block, start and size.