# User's Manual

TeamB - MPX

# Table of Contents

## R1 Commands

### Command 'getdate'

This command displays the current date to the user in the format: 'mm/dd/yy'.

Syntax:

$ *getdate*

### Command 'gettime'

This command displays the current time to the user in the format: 'hh:mm:ss'.

Syntax:

$ *gettime*

### Command 'help'

This command displays all possible commands and a simple description of their actions.

Syntax:

$ *help*

**Command 'setdate'**

This command allows the user to set the current date of the operating system. If an invalid date is entered, an error message is displayed to the user that reiterates the correct format. If no date is entered, an error message is displayed informing the user to enter the date directly after the command.

**mm** = desired month

**dd** = desired day

**yy** = desired year (assumed to be 21st century; date will be set to '20yy')

Syntax:

$ *setdate **mm/dd/yy***

**Command 'settime'**

This command allows the user to set the current time of the operating system. If an invalid time is entered, an error message is displayed to the user that reiterates the correct format. If no time is entered, an error message is displayed informing the user to enter the time directly after the command.

**hh** = desired month

**mm** = desired day

**ss** = desired seconds

Syntax:

$ *settime **hh:mm:ss***

**Command 'shutdown'**

This command allows the user to shut down the operating system. A prompt will appear asking the user to confirm their action by typing 'yes', upon which the operating system will cease operations and shut down. Any other entry will result in canceling the shutdown.

Syntax:

$ *shutdown*

Are you sure you want to shutdown?? (yes/no): *yes*


**Command 'version'**

This command displays the current version of the operating system (MPX) and when it was last compiled.

Syntax:

$ *version*

## R2 Commands

### Command 'createpcb'

This command allows the user to create a new PCB

**name** = process name

**class** = user application or system process

**priority** = 0 (highest priority) - 9 (lowest priority)

Syntax:

$ *createpcb* **name class priority**


### Command 'deletepcb'

This command allows the user to delete an existing PCB

**name** = process name

Syntax:

$ *deletepcb* **name**


### Command 'blockpcb'

This command allows the user to put a PCB in the blocked state and queue

**name** = process name

Syntax:

$ *blockpcb* **name**

**Command 'unblockpcb'**

This command allows the user to unblock a PCB - put a PCB in the ready state and queue

**name** = process name

Syntax:

$ un*blockpcb* **name**


**Command 'suspendpcb'**

This command allows the user to put a PCB in the suspended state and the corresponding

suspended queue

**name** = process name

Syntax:

$ *suspendpcb* **name**


**Command 'resumepcb'**

This command allows the user to put a PCB in the not-suspended state and the corresponding

not-suspended queue

**name** = process name

Syntax:

$ *resumepcb* **name**

**Command 'setpcbprio'**

This command allows the user to change the priority of an existing PCB

**name** = process name

**newpriority** = 0 (highest priority) - 9 (lowest priority)

Syntax:

$ *setpcbprio **name newpriority***


**Command 'showpcb'**

This command allows the user to display the attributes of a given PCB

**name** = process name

Syntax:

$ *showpcb **name***


**Command 'showreadypcbs'**

This command allows the user to display the attributes of all PCBs in the ready state/queue

Syntax:

$ *showreadypcbs*


**Command 'showblockedpcbs'**

This command allows the user to display the attributes of all PCBs in the blocked state/queue

Syntax:

$ *showblockedpcbs*

**Command 'showallpcbs'**

This command allows the user to display the attributes of all PCBs

Syntax:

$ *showallpcbs*