

# {nexTTeam} mpx module 1

User Manual

,

**{teaMMembers}**

*Teddy Hale*

*Jessica Hammersla*

*Mya Junkin*

*Philip Wenger*

## Table of Contents

<b>Overview</b>	<b>3</b>
<b>Beginning</b>	<b>3</b>
<b>Version</b>	<b>4</b>
<b>Date</b>	<b>4</b>
<b>Time</b>	<b>5</b>
<b>Shutdown</b>	<b>6</b>
<b>Help</b>	<b>6</b>
<b>Suspend/Resume</b>	<b>8</b>
<b>Set Priority</b>	<b>9</b>
<b>Show PCBs</b>	<b>10</b>
<b>Yield</b>	<b>13</b>
<b>Loadr3</b>	<b>13</b>

## Overview

Welcome to the Next Team Operating System (NTOS)! *Module One* introduces several essential features accessed through a menu-driven interface. This step in the NTOS development provides

- Date tools (Set Date/ Get Date)
- Time tools (Set Time/ Get Time)
- Help Info (Help)
- Version info (Version)
- (Shutdown)

The functionality provided with these features forms *NTOS Module One*. Use this manual to guide you through your first use! The next section addresses how to begin.

## Beginning

This section outlines how to boot NTOS and what to do upon booting, whenever the user is not yet doing anything.

### To Boot:

In a Linux environment, the user opens the terminal and changes the directory to where NTOS is stored. The user enters

```
$ Make
```

Then enters

```
$ qemu-system-i386 -nographic -kernel kernel.bin -s
```

The user is now running NTOS

### The Menu:

Upon booting, the screen is blank. The user is able to enter commands at the top of the screen. If, at any point, the user is unsure of what options are available to them, they may either enter

```
Help
```

Which will display a list of menu options, or they may refer to this manual. Whenever the user is ready to do something, they may enter a command of their choice.

## Version

This section explains the use of the command “version.” To begin, the user enters

```
version
```

This command returns the following:

```
VERSION: 1.0
```

This indicates that *NTOS Module One* is currently the version of NTOS in use.

## Date

This section explains the NTOS functionalities for setting and getting the system’s date.

### To Set the Date:

Setting the date changes the machine’s stored value for the current date. To begin setting the date, the user must enter

```
setDate
```

The user will then be prompted with the following message:

```
Enter date: mm/dd/yy
```

Beneath that, the user will enter the date in the format described by the prompt. For example, if the date to be entered is August 11th, 2019, the user will enter

```
08/11/19
```

If the user enters something in a format different than this, then nothing will happen and the user will be returned to the menu. If the correct format is used AND the date is valid, then the user will have set a new system date. The user will be returned to the menu.

### To Get the Date:

Getting the date lets the user see what value the machine has stored for the date. To begin getting the date, the user must enter

```
getDate
```

The system will then return the date in the following format:

```
mm/dd/yy
```

For example, if the stored values signified October 8th, 2006, the following would be returned:

```
11/08/06
```

The user will then be returned to the menu.

## Time

This section explains the NTOS functionalities for setting and getting the system's time.

### To Set the Time:

Setting the time changes the machine's stored value for the current time. To begin setting the time, the user must enter

```
setTime
```

The user will then be prompted with the following message:

```
Enter time: hh:mm:ss
```

Beneath that, the user will enter the time in the format described by the prompt (in military time). For example, if the time to be entered is 2:36 PM and 7 seconds, the user will enter

```
14:36:07
```

If the user enters something in a format different than this, then nothing will happen and the user will be returned to the menu. If the correct format is used AND the time is valid, then the user will have set a new system time. The user will be returned to the menu.

### To Get the Time:

Getting the time lets the user see what value the machine has stored for the time. To begin getting the time, the user must enter

```
getTime
```

The system will then return the time in the following format:

```
hh/mm/ss
```

For example, if the stored values signified 7:00 AM and 46 seconds, the following would be returned:

```
07:00:46
```

The user will then be returned to the menu.

## Shutdown

This section explains how to shutdown NTOS. To begin, the user will enter

```
shutdown
```

The user will then be prompted with

```
Are you sure you want to shut down? yes/no
```

If the user enters `yes`, the system will then shut down. If the user enters `no`, the user will be returned to the menu. If the user enters something else, the machine will prompt the user to

```
Please enter the word 'yes' or 'no'.
```

The user may then try entering `yes` or `no` again.

## Help

This section explains the use of the command “help.” To begin, the user enters

```
help
```

The user will then see

```
NAME
```

```
version - display current version of NTOS in use.
```

```
DETAIL DESCRIPTION
```

```
No further description
```

```
NAME
```

```
getTime - display current time of system.
```

```
DETAIL DESCRIPTION
```

```
Time will be displayed as hour:minute:second.
```

## NAME

setTime - change system's current time.

## DETAIL DESCRIPTION

Will prompt user to enter time as hh:mm:ss (i.e. hour:minute:second).

## NAME

getDate - display current date of system.

## DETAIL DESCRIPTION

Date will be displayed as month/day/year.

## NAME

setDate - change system's current date.

## DETAIL DESCRIPTION

Will prompt user to enter date as mm/dd/yy (i.e. month/day/year).

## NAME

Shutdown - shuts down NTOS.

## DETAIL DESCRIPTION

Will prompt user to confirm system shut down as yes/no.

## NAME

suspend - places PCB in suspended state.

## DETAIL DESCRIPTION

Will prompt user for a process name and will set to suspended state and move to the appropriate queue.

## NAME

resume - places PCB in not suspended state.

## DETAIL DESCRIPTION

Will prompt user for a process name and will set to not suspended state and move to the appropriate queue.

## NAME

setPriority - sets PCB priority.

## DETAIL DESCRIPTION

Will prompt user for a process name and ask for new priority and will set to new priority and move to the appropriate queue.

## NAME

showPCB - display info for a PCB.

## DETAIL DESCRIPTION

Will ask for a process name and display the process name, class, state, suspended status, and priority for a PCB.

NAME

showAllProcesses - display all PCBs.

DETAIL DESCRIPTION

Will display the process name, class, state, suspended status, and priority for all PCBs.

NAME

showReady - display all ready PCBs.

DETAIL DESCRIPTION

Will display the process name, class, state, suspended status, and priority for all ready queue PCBs.

NAME

showBlocked - display all blocked PCBs.

DETAIL DESCRIPTION

Will display the process name, class, state, suspended status, and priority for all blocked PCBs.

NAME

history - print command history.

DETAIL DESCRIPTION

Will display the user's ten previous commands.

The user is then returned to the menu.

## Suspend/Resume

This section explains the NTOS functionalities for suspending and resuming a process in the system.

### To Suspend a Process:

Suspending a process changes the PCB's stored value for its suspension state. To suspend a process, the user must enter

suspend

The user will then be prompted with the following message:

Enter the PCB name:



Beneath that, the user will enter the name of the process they wish to suspend.

If the user enters a name that does not match any process in the system, they will be given the following error.

```
There is no PCB by that name.
```

If the given name is valid, then the user will have set that process's suspension state to suspended. The user will be returned to the menu.

### To Resume:

Resuming a process changes the PCB's stored value for its suspension state. To resume a process, the user must enter

```
resume
```

The user will then be prompted with the following message:

```
Enter the PCB name:
```

Beneath that, the user will enter the name of the process they wish to resume.

If the user enters a name that does not match any process in the system, they will be given the following error.

```
There is no PCB by that name.
```

If the given name is valid, then the user will have set that process's suspension state to resume. The user will be returned to the menu.

## Set Priority

This section explains set a process's priority. To begin, the user will enter

```
setPriority
```

The user will then be prompted with

```
Enter the PCB name:
```

Beneath that, the user will enter the name of the process they wish to set priority for.

If the user enters a name that does not match any process in the system, they will be given the following error.

```
There is no PCB by that name.
```

If the given name is valid, then the user will be prompted again for the new priority number.

```
Enter the new priority for the PCB:
```

Beneath that, the user will enter the new priority for the PCB as an integer between 0 through 9 with 0 being of lowest priority. If a number outside of these bounds is entered the following error will be given:

```
The priority number is not valid
```

If the new number is valid a success message will, and the user will be returned to the menu.

## Show PCBs

### Show PCB:

To view one specific process's characteristics, the user must enter

```
showPCB
```

The user will then be prompted with the following message:

```
Enter the PCB name:
```

Beneath that, the user will enter the name of the process they wish to suspend.

If the user enters a name that does not match any process in the system, they will be given the following error.

```
There is no PCB by that name.
```

If the given name is valid, then the user will be shown that process's characteristics such as process name, class type (Application process or System process), state (Ready, Running, or Blocked), suspension state (Suspended or Not Suspended), and

priority number (an integer between 0-9 with 0 being lowest priority). This will be formatted as the example below:

```
Name: example
Class: System-Process
State: Ready
Suspend State: Not Suspended
Priority: 07
```

The user will then be returned to the menu.

### Show All Processes:

Showing all Processes shows the contents of all queues (Ready, Suspend-Ready, Blocked, and Suspend-Block) . To view all processes, the user must enter

```
showAllProcesses
```

The user will then view all processes within each queue in the format shown in the example below:

Ready Queue:

```
Name: dopey
Class: System-Process
State: Ready
Suspend State: Not Suspended
Priority: 07
```

Suspend Ready Queue:

```
Name: tootoo
Class: System-Process
State: Ready
Suspend State: Suspended
Priority: 02
```

Blocked Queue:

```
Name: example
Class: System-Process
State: Blocked
Suspend State: Not Suspended
```

```
Priority: 04
```

```
Suspend Blocked Queue:
```

The user will be returned to the menu.

### Show Ready Processes:

Showing Ready Processes displays the process characteristics of all processes in the Ready queue and Suspend-Ready queue. To show all ready processes, the user must enter

```
showReady
```

The user will then view all processes within each queue in the format shown in the example below:

```
Ready Queue:
```

```
Name: dopey
Class: System-Process
State: Ready
Suspend State: Not Suspended
Priority: 07
```

```
Suspend Ready Queue:
```

```
Name: tootoo
Class: System-Process
State: Ready
Suspend State: Suspended
Priority: 02
```

The user will be returned to the menu.

### Show Blocked Processes:

Showing Blocked Processes displays the process characteristics of all processes in the Blocked queue and Suspend-Blocked queue. To show all blocked processes, the user must enter

```
showBlocked
```

The user will then view all processes within each queue in the format shown in the example below:

Blocked Queue:

Name: example  
 Class: System-Process  
 State: Blocked  
 Suspend State: Not Suspended  
 Priority: 04

Suspend Blocked Queue:

The user will be returned to the menu.

## Yield

This section explains the use of the command “yield.” To begin, the user enters

version

This command then runs the next unsuspended ready process.

The user will be returned to the menu once all ready processes are finished.

## Loadr3

This section explains the use of the command “loadr3.” To begin, the user enters

loadr3

This command returns five consecutive success messages for suspending a PCB if successful.

This indicates that a PCB was created for each of the processes relating to module 3. They are loaded in as suspended-ready.

The user is immediately returned to the menu.