**potatOS**

(Version 0.3.0)

User Manual

Contents

[Overview of the OS 4](#_Toc432854542)

[Summary of Commands 5](#_Toc432854543)

[Summary of Commands (contd.) 6](#_Toc432854544)

[Detailed Description of Each Command 7](#_Toc432854545)

[version 7](#_Toc432854546)

[displaydate 7](#_Toc432854547)

[setdate 8](#_Toc432854548)

[restoredate 8](#_Toc432854549)

[directoryfiles 9](#_Toc432854550)

[createPCB 10](#_Toc432854551)

[deletePCB 11](#_Toc432854552)

[block 11](#_Toc432854553)

[unblock 12](#_Toc432854554)

[suspend 12](#_Toc432854555)

[resume 13](#_Toc432854556)

[setpriority 13](#_Toc432854557)

[showpcb 14](#_Toc432854558)

[showall 14](#_Toc432854559)

[showready 15](#_Toc432854560)

[showblocked 15](#_Toc432854561)

[sjf 16](#_Toc432854562)

[fifo 16](#_Toc432854563)

[stcf 17](#_Toc432854564)

[fpps 17](#_Toc432854565)

[rr 18](#_Toc432854566)

[mlfq 19](#_Toc432854567)

[ls 20](#_Toc432854568)

[help 21](#_Toc432854569)

[exit 21](#_Toc432854570)

[Possible Errors 22](#_Toc432854571)

[Possible Errors (contd.) 23](#_Toc432854572)

[Possible Errors (contd.) 24](#_Toc432854573)

[Index 25](#_Toc432854574)

# Overview of the OS

A brief description of the Operating System.

The potatOS operating system is a simple, barebones operating system (OS). As of this version of potatOS (v0.3.0) the OS can perform some basic actions as well as simulate a few process schedulers. However, in future updates, more functionality will be added. In its current state, the OS is merely an operating system simulator which runs on top of Microsoft’s Windows 10 OS. The purpose of this version is to set up a base user interface (UI) design as well as support some basic commands that a user will use (these commands will be discussed in the following sections).

Version 0.2.0 added functionality to be able to create and manipulate process control blocks (PCBs). With this implementation the OS has the foundation to simulate process scheduling which will be implemented in the future. Version 0.3.0 added functionality to simulate some common process schedulers. V0.3.0 also removed the user commands createpcb, deletepcb, block, and unblock.

# Summary of Commands

A brief description of the commands usable within the OS

In this version of potatOS (v0.2.0) there are 18 commands which the user can enter. The commands are shown in lower case but input is not case sensitive. The commands are as follows:

* version – Displays the current version of the OS on the screen.
* displaydate –Displays the date which is held by the OS.
* setdate – Displays a prompt to allow the user to set the date held by the OS.
* restoredate – Restores the date held by the OS to the current date.
* directoryfiles – Displays all of the files in the directory of the OS.
* createpcb—Allows the user to create a PCB.
* deletepcb—Deletes a PCB with a user specified name.
* block—Changes a user specified PCB’s state to blocked.
* unblock—Changes a user specified PCB’s state to ready.
* suspend—Changes a user specified PCB’s state to suspended.
* resume—Changes a user specified PCB’s state to not suspended.
* setpriority—Changes a user specified PCB’s priority.
* showpcb—Displays all information about a specified PCB.
* showall—Displays some information about all PCBs.
* showready—Displays some information about all PCBs in the ready state.
* showblocked—Displays some information about all PCBs in the blocked state.

# Summary of Commands (contd.)

* help – Displays a list of all of the valid commands and a brief description of each.
* sjf—Performs Shortest Job First process scheduler.
* fifo—Performs First In First Out process scheduler.
* stcf—Performs Shortest Time to Completion First process scheduler.
* fpps—Performs Fixed Priority Pre-Emptive Scheduler.
* rr—Performs Round Robin process scheduler.
* mlfq—Performs Multilevel Feedback Queue process scheduler.
* ls—Performs Lottery Scheduler.
* exit – Closes the OS.

# Detailed Description of Each Command

A more in-depth look at each command

## version

|  |  |
| --- | --- |
| Use | The “version” command is used if, for any reason, the user would want to know which version of the OS they are currently running. |
| Example |  |

## displaydate

|  |  |
| --- | --- |
| Use | The “displaydate” command is used if the user wants to know what the date is (by default the current date will be displayed, but the user can change this date by using the “setdate” command). |
| Example |  |

## setdate

|  |  |
| --- | --- |
| Syntax | *For entering a new day:* The number must be in the range of 1-31.  *For entering a new month:* The number must be in the range of 1-12.  *For entering a new year:* The number must be in the range of 1-9999.  (Disclaimer: there are no restrictions as to what the date can be set as other than these ranges [i.e. the user can set the date to 1/1/1]) |
| Use | The “setdate” command is used if the user wants to set the date held by the OS to a custom setting. After successfully entering a new date it will be the new date displayed by the OS unless the user enters “setdate” again or uses the “restoredate” command. |
| Example |  |
| Possible Errors | There are 3 different errors that could be displayed during this command:  “You must input a whole number between 1 and 31”,  “You must input a whole number between 1 and 12”, and  “You must input a whole number between 1 and 9999”.  Each of these errors simply let the user know that the value that they have input is not within the acceptable range. |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## restoredate

|  |  |
| --- | --- |
| Use | The “restoredate” command is used to restore the date held by the OS to the current date. |
| Example |  |

## directoryfiles

|  |  |
| --- | --- |
| Use | The “directoryfiles” command is used to display a list of all of the files that are in the OS’s directory. |
| Example |  |
| Possible Errors | There are two possible errors that this command can return:  “Not a directory” and  “No such file or directory”.  These two errors mean that the information that the OS is using to find the directory is invalid. |

## createPCB

|  |  |
| --- | --- |
| Syntax | *For entering a name:* The name must have at least one character.  *For entering the class:* The number must be 0 (for system) or 1 (for app).  *For entering a priority:* The number must be in the range of -127 to +128.  (Disclaimer: there are no restrictions as to what the name can be set as other than there cannot be a space between characters as it will cause the program to crash[i.e. ‘new pcb’ will crash the program but ‘$%fDF\*’ would be acceptable]) |
| Use | The “createpcb” function is used when the user wants to create and setup a new PCB. By default the PCB will be set as ready and not suspended. |
| Example |  |
| Possible Errors | There are 3 different errors that could be displayed during this command:  “\*ERROR: The input priority value was invalid. Try entering an integer between -127 and +128\*”,  “\*ERROR: The input class type was invalid. Try entering in either '1' for app or '0' for sys type\*”, and  “\*ERROR: The input name was not unique, try entering a new name\*”.  Each of these errors simply let the user know that the value that they have input is not acceptable and suggests what the user should try next time. |

## deletePCB

|  |  |
| --- | --- |
| Syntax | *For entering the name:* The name must match that of an existing PCB. |
| Use | The “deletepcb” function is used when the user wants to get rid of and delete a previously created PCB. |
| Example |  |
| Possible Errors | There is 1 error that could be displayed during this command:  “\*ERROR: No PCB found with that name\*”.  This error lets the user know that the name which they entered does not match any of the existing PCBs. This error can be displayed by several commands. |

## block

|  |  |
| --- | --- |
| Syntax | *For entering the name:* The name must match that of an existing PCB. |
| Use | The “block” function is used when the user wants to change the state of a PCB to blocked. |
| Example |  |
| Possible Errors | There is 1 error that could be displayed during this command:  “\*ERROR: No PCB found with that name\*”.  This error lets the user know that the name which they entered does not match any of the existing PCBs. |

## unblock

|  |  |
| --- | --- |
| Syntax | *For entering the name:* The name must match that of an existing PCB. |
| Use | The “unblock” function is used when the user wants to change the state of a PCB to unblocked. |
| Example |  |
| Possible Errors | There is 1 error that could be displayed during this command:  “\*ERROR: No PCB found with that name\*”.  This error lets the user know that the name which they entered does not match any of the existing PCBs. |

## suspend

|  |  |
| --- | --- |
| Syntax | *For entering the name:* The name must match that of an existing PCB. |
| Use | The “suspend” function is used when the user wants to change the state of a PCB to suspended. |
| Example |  |
| Possible Errors | There is 1 error that could be displayed during this command:  “\*ERROR: No PCB found with that name\*”.  This error lets the user know that the name which they entered does not match any of the existing PCBs. |

## resume

|  |  |
| --- | --- |
| Syntax | *For entering the name:* The name must match that of an existing PCB. |
| Use | The “resume” function is used when the user wants to change the state of a PCB to ready. |
| Example |  |
| Possible Errors | There is 1 error that could be displayed during this command:  “\*ERROR: No PCB found with that name\*”.  This error lets the user know that the name which they entered does not match any of the existing PCBs. |

## setpriority

|  |  |
| --- | --- |
| Syntax | *For entering the name:* The name must match that of an existing PCB.  *For entering the priority:* The number must be in the range of -127 to +128. |
| Use | The “setpriority” function is used when the user wants to change the priority of a PCB. |
| Example |  |
| Possible Errors | There are 2 errors that could be displayed during this command:  “\*ERROR: No PCB found with that name\*” and  “\*ERROR: The input priority value is invalid\*”.  These errors lets the user know that the name which they entered does not match any of the existing PCBs or that the user has entered a priority value outside of the acceptable range. |

## showpcb

|  |  |
| --- | --- |
| Syntax | *For entering the name:* The name must match that of an existing PCB. |
| Use | The “showpcb” function is used when the user wants to see the attributes/information of a PCB. |
| Example |  |
| Possible Errors | There is 1 error that could be displayed during this command:  “\*ERROR: No PCB found with that name\*”.  This error lets the user know that the name which they entered does not match any of the existing PCBs. |

## showall

|  |  |
| --- | --- |
| Use | The “showall” function is used when the user wants to see some of the attributes/information of all PCBs. |
| Example |  |
| Possible Errors | There is 1 error that could be displayed during this command:  “\*ERROR: There are currently no PCBs to show information\*”.  This error lets the user know that no PCBs have been created so no information can be displayed. |

## showready

|  |  |
| --- | --- |
| Use | The “showready” function is used when the user wants to see some of the attributes/information of all PCBs in the ready state. |
| Example |  |
| Possible Errors | There is 1 error that could be displayed during this command:  “\*ERROR: There are no PCBs in the ready state\*”.  This error lets the user know that no PCBs are in the ready state to show information about. |

## showblocked

|  |  |
| --- | --- |
| Use | The “showblocked” function is used when the user wants to see some of the attributes/information of all PCBs in the blocked state. |
| Example |  |
| Possible Errors | There is 1 error that could be displayed during this command:  “\*ERROR: There are no PCBs in the blocked state\*”.  This error lets the user know that no PCBs are in the blocked state to show information about. |

## sjf

|  |  |
| --- | --- |
| Syntax | *For entering the file name:* The name must match that of an existing file in the working directory. |
| Use | The “sjf” function is used when the user wants to simulate the Shortest Job First process scheduler. |
| Example |  |
| Possible Errors | There are 2 errors that could be displayed during this command:  “\*ERROR: Unable to open file\*” &  “\*ERROR: File is already open”.  These errors let the user know that either the entered filename is invalid/incorrect or that the file is already open. |

## fifo

|  |  |
| --- | --- |
| Syntax | *For entering the file name:* The name must match that of an existing file in the working directory. |
| Use | The “fifo” function is used when the user wants to simulate the First In First Out process scheduler. |
| Example |  |
| Possible Errors | There are 2 errors that could be displayed during this command:  “\*ERROR: Unable to open file\*” &  “\*ERROR: File is already open”.  These errors let the user know that either the entered filename is invalid/incorrect or that the file is already open. |

## stcf

|  |  |
| --- | --- |
| Syntax | *For entering the file name:* The name must match that of an existing file in the working directory. |
| Use | The “stcf” function is used when the user wants to simulate the Shortest Time to Completion First process scheduler. |
| Example |  |
| Possible Errors | There are 2 errors that could be displayed during this command:  “\*ERROR: Unable to open file\*” &  “\*ERROR: File is already open”.  These errors let the user know that either the entered filename is invalid/incorrect or that the file is already open. |

## fpps

|  |  |
| --- | --- |
| Syntax | *For entering the file name:* The name must match that of an existing file in the working directory. |
| Use | The “fpps” function is used when the user wants to simulate the Fixed Priority Pre-Emptive Scheduler. |
| Example |  |
| Possible Errors | There are 2 errors that could be displayed during this command:  “\*ERROR: Unable to open file\*” &  “\*ERROR: File is already open”.  These errors let the user know that either the entered filename is invalid/incorrect or that the file is already open. |

## rr

|  |  |
| --- | --- |
| Syntax | *For entering the file name:* The name must match that of an existing file in the working directory.  *For entering the time slice duration:* The number must be an integer within the acceptable range displayed. |
| Use | The “rr” function is used when the user wants to simulate the Round Robin scheduler. |
| Example |  |
| Possible Errors | There are 3 errors that could be displayed during this command:  “\*ERROR: Unable to open file\*”,  “\*ERROR: File is already open”, &  “\*ERROR: Input number invalid, try again”.  These errors let the user know that either the entered filename is invalid/incorrect, that the file is already open, or that the number input for the time slice was invalid. |

## mlfq

|  |  |
| --- | --- |
| Syntax | *For entering the file name:* The name must match that of an existing file in the working directory.  *For entering the time slice(s) duration:* The number must be an integer within the acceptable range displayed.  *For entering the number of priority levels:* The number must be an integer within the acceptable range displayed |
| Use | The “rr” function is used when the user wants to simulate the Round Robin scheduler. |
| Example |  |
| Possible Errors | There are 3 errors that could be displayed during this command:  “\*ERROR: Unable to open file\*”,  “\*ERROR: File is already open”, &  “\*ERROR: Input number invalid, try again”.  These errors let the user know that either the entered filename is invalid/incorrect, that the file is already open, or that the number input for the time slice/number of priorities was invalid. |

## ls

|  |  |
| --- | --- |
| Syntax | *For entering the file name:* The name must match that of an existing file in the working directory.  *For entering the time slice duration:* The number must be an integer within the acceptable range displayed.  *For entering the number of tickets:* The number must be an integer greater than the minimum displayed. |
| Use | The “rr” function is used when the user wants to simulate the Round Robin scheduler. |
| Example |  |
| Possible Errors | There are 3 errors that could be displayed during this command:  “\*ERROR: Unable to open file\*”,  “\*ERROR: File is already open”, &  “\*ERROR: Input number invalid, try again”.  These errors let the user know that either the entered filename is invalid/incorrect, that the file is already open, or that the number input for the time slice/number of tickets was invalid. |

## help

|  |  |
| --- | --- |
| Use | The “help” command is used to display a list of all valid commands along with a brief description for each. |
| Example |  |

## exit

|  |  |
| --- | --- |
| Use | The “exit” command is used in order to exit out of the OS. Once entered the user will be asked if they are sure that they wish to exit. From there they can enter ‘y’ or ‘yes’ to close the program, or they can enter ‘n’ or ‘no’ to return back to the command prompt. If they enter ‘y’ or ‘yes’, they will be required to press the enter key to completely close the window. |
| Example |  |
| Possible Errors | There is one possible error within the “exit” command:  “ERROR: You have entered an invalid response.”  This error is displayed when the user enters in a value other than those acceptable (‘y’, ‘yes’, ‘n’, and ‘no’). |

# Possible Errors

|  |  |
| --- | --- |
| **Command** | **Error Description** |
| setdate | *“You must input a whole number between 1 and 31”*  This error is shown when the user has input a value that is not in the range of 1 to 31. Since there are no months which have a day that is less than 1 or greater than 31, this error is shown instead of accepting a value outside of the acceptable range. |
|  | *“You must input a whole number between 1 and 12”*  This error is shown when the user has input a value that is not in the range of 1 to 12. Since there are only 12 months, this error is shown instead of accepting a value outside of the acceptable range. |
|  | *“You must input a whole number between 1 and 9999”*  This error is shown when the user has input a value that is not in the range of 1 to 9999. Since the OS only accepts a year value of up to 4 digits, this error is shown instead of accepting a value outside of the acceptable range. |
| directoryfiles | *“Not a directory”*  This error is shown when the information that the OS is using to find the location of the directory is not actually valid directory information. |
|  | *“No such file or directory”*  This error is shown when the information that the OS is using to find the location of the directory does not actually lead to a directory. |
| exit | *“ERROR: You have entered an invalid response.”*  This error is shown when the user has entered an unacceptable value when they are asked if they are sure that they wish to exit the program. |
| showall | *“\*ERROR: There are currently no PCBs to show information\*”*  This error is shown when no PCBs exist. This can be either because the user has not created any or that they have all been deleted. |
| showready | *“\*ERROR: There are currently no PCBs in the ready state\*”*  This error is shown when no PCBs are in the ready state. |

# Possible Errors (contd.)

|  |  |
| --- | --- |
| **Command** | **Error Description** |
| showpcb  setpriority  block  unblock  suspend  resume  deletepcb | *“\*ERROR: No PCB found with that name\*”.*  This error is shown when the user has entered in a name which does not match the name of any of the existing PCBs. |
| showblocked | *“\*ERROR: There are currently no PCBs in the blocked state\*”*  This error is shown when no PCBs are in the blocked state. |
|  |  |
| setpriority | *“\*ERROR: The input priority value is invalid\*”*  This error is shown when the user has input a priority value that is either less than -127 or greater than 128. |
| createpcb | “\*ERROR: The input priority value was invalid. Try entering an integer between -127 and +128\*”  This error is shown when the user has input a priority value that is either less than -127 or greater than 128.  “\*ERROR: The input class type was invalid. Try entering in either '1' for app or 0' for sys type\*”  This error is shown when the user has input a class value that is not a 1 or 0.  “\*ERROR: The input name was not unique, try entering a new name\*”  This error is shown when the user has entered a name that is already in use by another PCB. |
| ls  mlfq  rr | “\*ERROR: Invalid input, try again”  This error is shown when the user enters a number for a time slice, number of tickets, or number of priority levels that the prompt displayed forbids. |
| sjf  fifo  stcf  fpps  rr  mlfq  ls | “\*ERROR: Unable to open file\*” |

# Possible Errors (contd.)

|  |  |
| --- | --- |
| **Command** | **Error Description** |
| sjf | “\*ERROR: File is already open”  This error is shown when the user enters a file name to a file that is already open. |

# Index

block, 5, 11, 23

case sensitive, 5

createpcb, 5, 10, 23

deletepcb, 5, 11, 23

directoryfiles, 5, 9, 22

displaydate, 5, 7

exit, 6, 21, 22

fifo, 2, 6, 23

fpps, 2, 6, 23

help, 6, 21

ls, 2, 6, 23

mlfq, 2, 6, 23

restoredate, 5, 8

resume, 5, 13, 23

rr, 2, 6, 23

setdate, 5, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 22, 23

setpriority, 5, 13, 23

showall, 5, 14, 22

showblocked, 5, 15, 23

showpcb, 5, 14, 16, 17, 18, 19, 20, 23

showready, 5, 15, 22

sjf, 2, 6, 23, 24

stcf, 2, 6, 23

suspend, 5, 12, 23

unblock, 5, 12, 23

version, 5, 7