Technical MANUAL

Ballenger Operating System (BOS)

Table of Contents

[MAIN.CS 2](#_Toc8233119)

[RAM.CS 2](#_Toc8233120)

[CPU.CS 3](#_Toc8233121)

[MACHINE.CS 3](#_Toc8233122)

[FREELIST.CS 4](#_Toc8233123)

[NODE.CS 4](#_Toc8233124)

[UnitTest1.CS 4](#_Toc8233125)

## 

\*\*BOS is written in C# using visual studio\*\*

## MAIN.CS

Global variables

* string CurrentDate
* ArrayList versionCommands
* ArrayList dateCommands
* ArrayList directoryCommands
* ArrayList historyCommands
* ArrayList runCommands
* ArrayList aliasCommands
* ArrayList exitCommands
* ArrayList helpCommands
* ArrayList executeCommands

Methods

* static void Main(string[] args)
* static public string Version()
* static public void Date()
* static public string ChangeDate(string newDate)
* static public string[] Dir(string path)
* static public void History(ArrayList array)
* static public string[] Run(string fileLocation)
* static public void Alias(string commandName, string newname)
* static public void Help(string response)

## RAM.CS

Global variables

* string[] ram
* FreeList freelist

Methods

* Int findFreeSpace(int sizeNeeded)
* Int loadProcess(string[] instructions, int startingLocation)
* String getValue(int location)
* Void setValue(string value, int location)

## CPU.CS

Global variables

* String ax
* String bx
* String cx
* String dx
* Int pc
* Int a

Methods

* Void setAX(string value)
* Void setBX(string value)
* Void setCX(string value)
* Void setDX(string value)
* Void setPC(int value)
* Void incrementPC()
* Void setA(int value)
* String getAX()
* String getBX()
* String getCX()
* String getDX()
* Int getPC()
* Int getA()

## MACHINE.CS

Global variables

* OSProject.RAM ram
* OSProject.CPU cpu
* Dictionary<string,int> processtable
* Int processID

Methods

* Void addProcessToTable(int ramlocation)
* String[] buildPCB()
* Int loadForExecution(string fileName)
* Void executeInstructions(int[] processLengths)
* Void loadPCB(int pcb)
* Void savePCB(int pcbStart)
* String tick(string instruction)
* String commaToken(string instruction)
* String endToken(string instruction)
* String singleToken(string instruction)
* Void statusDisplay(int start, int end)
* String valueToken(string token)
* String checkRam(int location)
* String checkAX()

## FREELIST.CS

Global variables

* Node head
* Node current
* Node tail

Methods

* Int size()
* Void append(int start, int end)
* Void remove()
* Int changeCurrentStart(int start)
* Int changeCurrentEnd(int end)
* Void currentToNext()
* Int getHeadStart()
* Int getHeadEnd()
* Int getCurrentStart()
* Int getCurrentEnd()

## NODE.CS

Global variables

* Int start
* Int end
* Node prev
* Node next

Methods

* Int start
* Int end
* Node prev
* Node next

## UnitTest1.CS

Tests

* Test\_VersionMethod()
* Test\_ChangeDateMethod()
* Test\_DirMethod()
* Test\_RunMethod()
* Test\_SetValueMethod()
* Test\_SetAXMethod()
* Test\_SetBXMethod()
* Test\_SetCXMethod()
* Test\_SetDXMethod()
* Test\_SetPCMethod()
* Test\_incrementPCMethod()
* Test\_SetAMethod()
* Test\_tickMethod()
* Test\_commaTokenMethod()
* Test\_endTokenMethod()
* Test\_singleTokenMethod()
* Test\_appendMethod()
* Test\_sizeMethod()
* Test\_loadMethod()
* Test\_loadForExecution()