Overview

The purpose of this program is to count characters supplied by a data file as an argument when the program is run and then display output containing three points of data: the character, it ASCII code, and the frequency of the character in the file.

### PROCESSING LOGIC

The Main method of the Program class is executed first. It takes an argument, tests for additional arguments, and outputs a message if more than one argument is supplied. Then it performs

else HandleFile(ReadFile(args));

passing control to HandleFile, providing the output of ReadFile as an argument.

ReadFile performs the following operations:

private void ReadFile(string[] args)

if args is not null

if the file exists

try to make a string of the full path

use StreamReader to open a stream.

Call ProcessFile.

Catch the exception

Catch the exception

HandleFile performs the following operations:

private void HandleFile(char[] chars)

create an instance of a linked list of type CharacterFrequency called frequencyList

initiate a foreach control loop on the character array

instantiate a CharacterFrequency object, cf, passing the contructor the character we’d like to test

declare a decision statement checking the linked list for whether it .Contains() a new CharacterFrequency object, aCharacter

if it does, call IncrementFrequency() on instance cf of CharacterFrequency

else, add the item to the end of the linked list

for each of the items in the linked list,

use the ToString() override to provide the requested output

OTHER CLASSES

The main program flow must be supplemented with the flow of subroutines/methods/functions that are called from the main program.

Any specific algorithms to be used should be stated or referenced.

### DATA (INPUT/OUTPUT)

The program accepts a single input in the form of a text file.

The text file is entirely printable ASCII.

Each character in the text file must be found on the ASCII table.

Values of each element are their ASCII values

### COMPONENTS (SOURCE CODE NAMES, CLASSES, METHODS)

Program class- entry point. Main method calls HandleFile, providing the output from ReadFile as an argument.

UnsupportedException class : Esception

Contains three constructors, none of which require implementation due to the use of the switch/case in UserInterface:

public UnsupportedArgsException()

public UnsupportedArgsException(string message)

: base(message)

public UnsupportedArgsException(string message, Exception inner)

: base(message, inner)

ControlManager class: its methods are named and described above.

UserInterface class:

public string Output { get; set; } Stores the Output messages as strings for retrieval.

private string Convert(CharacterFrequency[] freq)

public void ExceptionMessageOutput(Type ex)

public void DisplayOutput(CharacterFrequency[] freqArray) // these are described above.

Include UML class diagram showing the properties and methods of the class.

Describe the properties of the class and any constraints upon their value.

Indicate where the algorithms defined in processing logic are implemented.

### TESTING

Present one or more named scenarios including the input data, expected output and the success criteria desired that will be utilized to test the application. The testing plan should be repeatable so discuss any ‘clean up’ that might be required to do so.

Scenario 1 – output file arg test

Steps to test

Supply an output file argument in addition to the input file when the program is run.

Expected reaction: output reporting an unsupported argument was supplied.

Expected results is a message saying “File writing is unsupported."

Scenario 2 – FileNotFoundException

Steps to test

Move the text file to be supplied as input out of the directory containing the executable being run.

Expected result is that the program throws a FileNotFoundException.

Scenario 3 – IndexOutOfRangeException

Steps to test

Fail to provide any arguments when the program is run.

Expected results is a message indicating the index was out of range. (This happens because args[] is empty).