

Assignment Prefix: Lab04

Due Date: Friday, Feb. 10th @ 11:59pm

Points: 100

This is an individual assignment.

Restrictions: you cannot use any methods from the Java Array(s) class to copy an array, check for equality, or otherwise manipulate an array. You must write the Java code to perform these functions.

In completing the find method, you will need to use the Java File class. You may use any of the methods that are available in the Java File class in developing your algorithm.

Create a NetBeans project named Lab04 and ensure it is saved to a location like desktop or your flash drive. In the project you will do the following:

In this assignment you are to create a Recursion Class that will implement the following recursive algorithms:

- Implement a recursive algorithm to compute the integer part of the base-two logarithm of n using only addition and integer division as described in problem C-5.11 on page 221 of the textbook.
- Implement a recursive algorithm to compute the product of two positive integers, m and n, using only addition and subtraction as described in problem C-5.13 on page 221 of the textbook.
- Implement Isabel's technique for summing the values in an array of n integers as described in problem C-5.24 on page 223 of the textbook.
- Implement a recursive method with calling signature find(path, filename) that reports all entries of the file system rooted at the given path having the given filename as described in problem P-5.27 on page 223 of the textbook.

Create a Recursion_Client Class that will fully test each of the recursive methods that you created above in an interactive fashion. This class should:

- Use a JOptionPane to ask the user which algorithm they wish to test.
- Use a JOptionPane to ask the user for any parameters/values needed for the test.
 - For Isabel's technique the parameter should be the path and filename of a text file that contains a comma separated list of integers values that will make up the array.

- When you read in the values from the file you should skip any values that are not integers.
- If the user enters a file that does not exist your program should inform the user that the file does not exist and ask the user to enter a correct filename or quit back to algorithm selection.
- Output for each test should be shown in the system console and must include:
 - The name of the algorithm being tested.
 - The parameters entered by the user
 - Values used if different from parameters (e.g. the contents of the array read in for Isabel's technique).
 - Error messages give to the user (e.g. file not found)
 - The results of the test.
 - Note that the JOptionPane interactions will not be captured in the system console.
- At the end of a test the user should be given the options of running another test or exiting the program.

You may want to review Chapter 11 from Java Illuminated. A scanned copy of this chapter is available on Blackboard in Course Content -> Textbook Scans.

Things to turn in:

- Open a Microsoft Word document
- Copy and Paste the source code of the **Recursion Class** (make sure to use *Ctrl + A* to select all the source code of the program, *Ctrl + C* to copy, and *Ctrl + V* to paste.).
- Copy and Paste the source code of the **Recursion_Client** Class
- Copy and paste the output of the client program
- Next, zip the Project folder.
- Finally on blackboard, submit both your Word document and project zipped file.