AIST 2120
Programming Assignment 6

INTRODUCTION

This assignment presents another opportunity to connect some dots. Chapter 15 introduced us to the wonderful world of working with PDF documents. We learned about JSON files in Chapter 16 and keeping time in Chapter 17. Let's put some of those pieces together.

ASSIGNMENT & DISCUSSION

Your mission in this homework assignment is to practice working with PDF documents, creating JSON files, and keeping time.

IMPORTANT: A sizeable portion of this Programming Assignment looks suspiciously like Lab Exercise 15. ***Reuse your working Lab 15 code for this assignment! You should then only have to add the time keeping and JSON work.***

Of note, you'll be working with three seminal papers in the computing world. Alan Turing authored two of the papers you'll see. Turing is considered the Father of Computer Science. His 1936 paper introduced such topics as Turing Machines and Universal Turing Machines. His 1950 paper introduced the Turing Test. Claude Shannon, the Father of Information Theory, contributed the third paper, which describes the basic logic functions at the heart of all electronic digital computer designs. All three are worth adding to your library and reading!

You are to write a program to *'discover'* the PDF documents in a folder of a variety of filetypes. You will disregard all non-PDF documents. You will apply encryption to the PDF documents.

To make this interesting, you will keep tabs on *how long it takes to process (read, encrypt, write) each PDF document*. You'll also *create a JSON file that keeps track of each PDF and the length of time it took to process it.* Sounds super fun, doesn't it?

Tasks

- **IMPORTANT**: Download the provided **Programming Assignment 6 ZIP** archive. Open the archive and save the **PA6** folder to your desktop. This will be your working directory.
- **IMPORTANT**: Use relative paths in your source code; i.e., *do not use absolute paths*.
- Note the folder contents:
    - *00 – READ_ME.txt* file,
    - *Source* subfolder, and
    - The Source folder contains another **00 - READ_ME.txt** file along with other .**docx, .pdf, .txt**, and **.xlsx** files.
    - Don't move, remove or alter any these files.
    - The ***lastN_Destination*** folder only contains a 00 – READ_ME.txt file. This folder will be set up when you run your code. A function exists in your example file for setting up the destination folder. It will also hold your processed files. In other words, you will write files to this folder.
- Write your program to accomplish the following steps:
    (1) Search the Source folder
    (2) Write the filenames of the files found in the **Source** folder to the screen (See sample output).
    (3) Copy each **.pdf** file (only pdf) to the ***lastN_Destination*** folder (Apply encryption to each .pdf file with password "enigma")
        (a) Rename the newly encrypted PDF files using the original filename with a prefix of "**encrypted_**". For example: A file named **abc.pdf** should have the new filename of **encrypted_abc.pdf**.

(4) Keep track of how long it took to process each PDF; steps 3, 4 and 5. (**Hint**: Refer to Chapter 17 treatment of the time module.)
(5) Create a dictionary that has key-value pairs of the original processed filename (key) and the time it took to process the original file (value). Round the time value to two digits after the decimal point.
    (a) for example: "filename": 1.27.
    (b) Convert the dictionary to JSON and write it to a JSON file (**lastN_time_span.json**) in the **lastN_Destination** folder
(6) Output the filenames of the processed PDF files placed in the **lastN_Destination** folder to the screen (See sampleoutput below).

Notes

- The PDF work in this assignment is rather straightforward. Refer to *Encrypting PDFs* on pages 354 – 355 in our textbook. **Remember the above note on Lab 15!**
- Refer to Chapters 9 and 10 for file operations that will help meet the requirements of this assignment.
- **Hint**: You may want to walk through all files in the **Source** folder to determine which files to process. See *Walking a Directory* Tree on pages 235 – 237 of our text or in os.listdir() page 213.
- **Hint**: An interesting point is that PDF files always '***end with'*** .pdf. That's pretty obvious, right? That could be handy to determine which **Source** folder files to work with and which to ignore…
- **Hint**: Break down your project into phases. I recommend something like the following:
    1. Get your program to search all files in a folder for PDF documents (.pdf)
    2. Get your program to add encryption to PDF documents
    3. Get your program to read from the Source folder and write to the Destination folder
    4. Get your program to time how long it takes to process a PDF document
    5. Get your program to store the PDF filename and processing time in a dictionary
    6. Have your program convert the finished dictionary to a JSON file (.json)
    7. Write the JSON file out to the correct location

REQUIREMENTS

Application

- Write your program in Python 3.
- You are free to develop your code in any environment you choose, but I will run your code in IDLE's File Editor. If you choose to write your code in another environment, I strongly encourage you to test your code in IDLE's File Editor prior to submission.For grading purposes, if your code doesn't run in IDLE's File Editor, it doesn't run.
- The PA6 folder is included in the provided PA6 ZIP archive.
- Name your destination folder **lastN_Destination**
- Name your source code file **lastN_firstN_pdf_encrypt_plus.py**.
- Name your JSON file **lastN_time_file.json**.

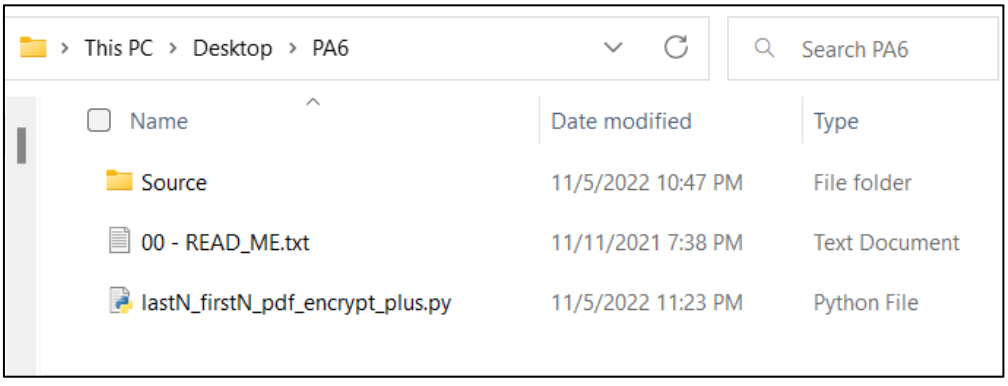
SUBMISSION

- Submit your source code files via D2L. I will not accept email or hard copy submissions.
- Ensure you retain a complete copy of your program source code file(s).

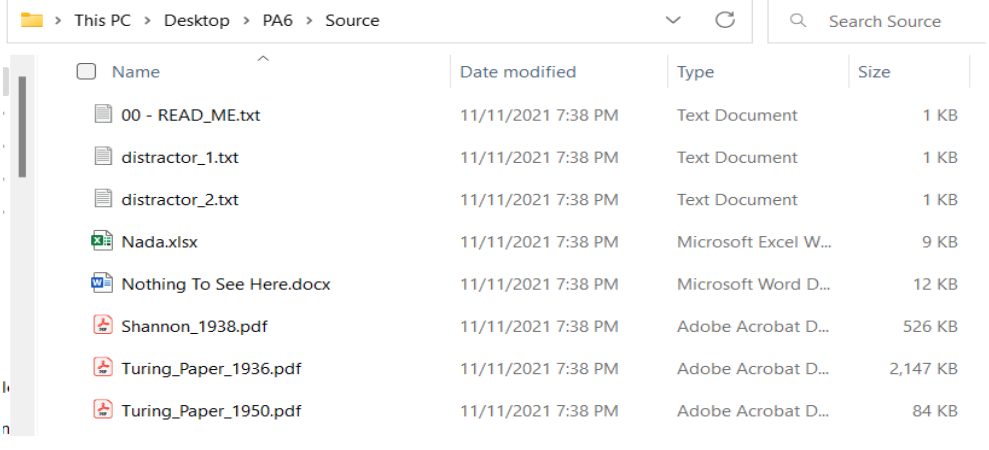
Due Date: Per D2L instructions.
Late Penalty: See Syllabus

STYLE GUIDE: **File headers**. Include a header in the below format at the top of all .py source code files.#
U.R. Name
AIST 2120C, Programming Assignment 7
xx/xx/xxxx
lastN_firstN_pdf_encrypt_plus.py

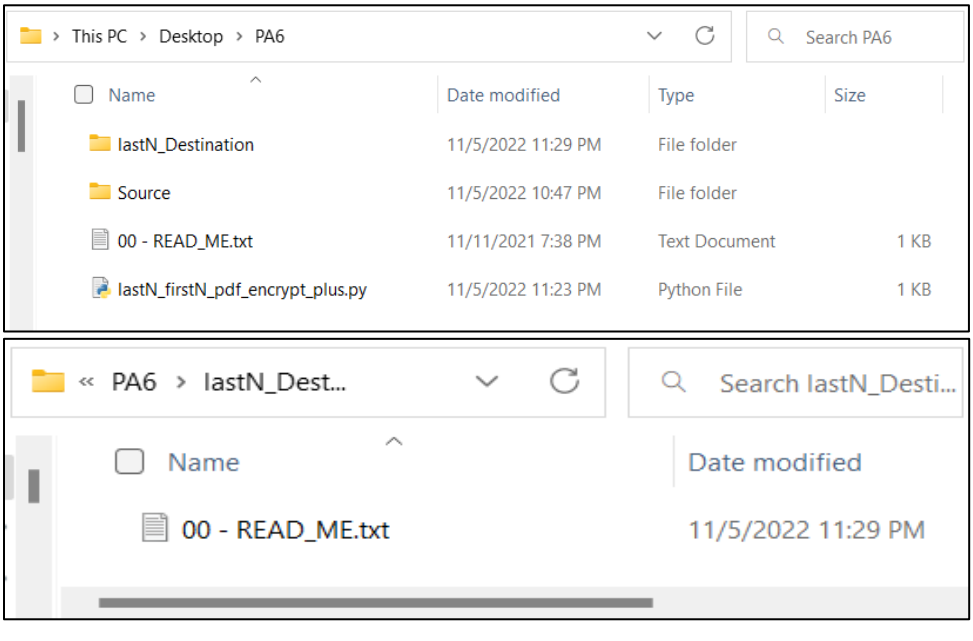
Provided Project Folder Contents



Provided Source Folder Contents



Destination Folder contents after running the provided code



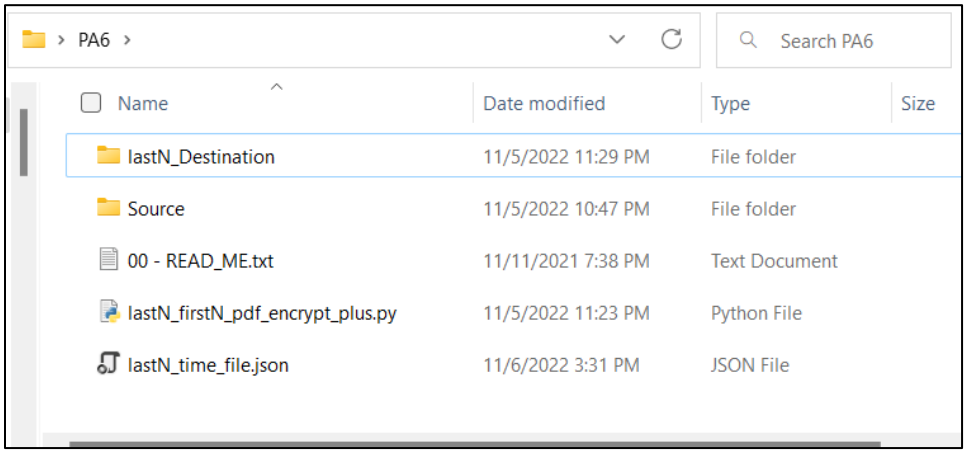
Example Screen Output

```
          ------------------------------------------------
          ----------          AIST 2120C          ----------
          ----------          Jacob Cox           ----------
          ----------   Programming Assignment 7   ----------
          ----------      PDF Encryptinator       ----------
          ------------------------------------------------

Processing folder: .\Source\

Visible files found:
 -  00 - READ_ME.txt
 -  distractor_1.txt
 -  distractor_2.txt
 -  Nada.xlsx
 -  Nothing To See Here.docx
 -  Shannon_1938.pdf
 -  Turing_Paper_1936.pdf
 -  Turing_Paper_1950.pdf

        ------------------------------

Processed files location: .\Destination\

Processed files:
 - encrypted_Shannon_1938.pdf
 - encrypted_Turing_Paper_1936.pdf
 - encrypted_Turing_Paper_1950.pdf
Created file: time_span_JHC.json


*** Mission Complete ***

          ------------------------------------------------
          ----------   Programming Assignment 7   ----------
          ----------      PDF Encryptinator       ----------
          ----------           Complete           ----------
          ------------------------------------------------
```

Example lastN_time_span.json File Contents

```
1  {"Shannon_1938.pdf": 0.6,
   "Turing_Paper_1936.pdf": 3.41,
   "Turing_Paper_1950.pdf": 0.08}
```

Example Project Folder Post Run Contents



Example Destination Folder Post Run Contents