## Visvesvaraya Technological University

## Belagavi-590 018, Karnataka



A MINI PROJECT REPORT ON

### “Document Processing”

**Mini-Project Report submitted in fulfillment of the requirement for the 6th Semester File Structures Laboratory with Mini-Project**

**[17ISL68]**

## Bachelor of Engineering

**In**

### Information Science and Engineering

**Submitted by**

**RUTHVIK L[1JT17IS035]**

Under the Guidance of

**Mr.Vadiraja A**

Asst.Professor,Department of ISE



**Department of Information Science and Engineering**

**Jyothy Institute of Technology,**

**Tataguni, Bengaluru-56008**

**Jyothy Institute of Technology,**

**Department of Information Science and Engineering**

**Tataguni, Bengaluru-560082**



**CERTIFICATE**

Certified that the mini project work entitled **“Text File And Image Compression”** carried out by **RUTHVIK L[1JT17IS035]** Bonafede student of Jyothy Institute of Technology, in partial fulfillment for the award of **Bachelor of Engineering** in **Information Science and Engineering** Department of the **Visvesvaraya Technological University, Belagavi** during the year **2020-2021**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

**Mr. Vadiraja A** **Dr. Harshvardhan Tiwari**

Guide, Asst, Professor Associate Professor and HoD

Dept. Of ISE Dept. OF ISE

External Viva Examiner Signature with Date :





### ACKNOWLEDGMENT

The satisfaction that accompanies the successful completion of my project Would be incomplete without mentioning the people who made it possible, I am thankful to **Dr. Gopalakrishna K,** Principal, JIT, Bangalore for Being kind enough to provide us an opportunity to work on a project in this Institution

I am thankful to **Dr. Harshvardhan Tiwari ,** HoD, Dept ofISE for his co-operation and encouragement at all moments of our approach and Whose constant guidance and encouragement crowns all the efforts with Success.

I would greatly mention enthusiastic influence provided by **Mr. Vadiraja A,** Asst. Professor, Dept. of ISE for his ideas and co-operation Showed on us during our venture and making this project a great success.

Finally, it’s pleasure and happiness to the friendly co-operation showed by all The staff members of Information Science Department, JIT.

**RUTHVIK L**

**1JT17IS035**

**ABSTRACT**

This project is done using Spyder IDE for python programming language and tkinter package to create the front-end. The tittle of the project is “Document Processing”. In this project we will be accessing all the files in a folder and searching and replacing the texts as per user request either in the entire document or only in a specified line.

.

### TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **SERIAL NO.** | **DESCRIPTION** | **PAGE NO.** |
|  | **Chapter 1** |  |
| 1 | **Introduction** | 1 – 3 |
| 1.1 | Introduction to File Structures | 1 |
| 1.2 | Introduction to File System | 2 |
| 1.3 | Data Compression | 3 |
|  |  |  |
|  | **Chapter 2** |  |
| 2 | Requirement analysis and design | 4-5 |
| 2.1 | Domain Understanding | 4 |
| 2.2 | Classification of Requirements | 4 |
| 2.3 | System Analysis | 5 |
|  |  |  |
|  |  |  |
|  | **Chapter 3** |  |
| 3 | **Implementation** | 6-8 |
|  |  |  |
|  | **Chapter 4** |  |
| 4 | **Result and Analysis** | 9-14 |
|  |  |  |

**CHAPTER 1 INTRODUCTION**

### INTRODUCTION

* 1. **Introduction to File Structures**

In simple terms, a file is a collection of data stored on mass storage (e.g., disk or tape).But there is one important distinction that must be made at the outset when discussing file structures. And that is the difference between the logical and physical organization of the data.

On the whole a file structure will specify the logical structure of the data, that is the relationships that will exist between data items independently of the way in which these relationships may actually be realized within any computer. It is this logical aspect that we will concentrate on. The physical organization is much more concerned with optimizing the use of the storage medium when a particular logical structure is stored on, or in it. Typically for every unit of physical store there will be a number of units of the logical structure (probably records) to be stored in it.

For example, if we were to store a tree structure on a magnetic disk, the physical organization would be concerned with the best way of packing the nodes of the tree on the disk given the access characteristics of the disk.

Like all subjects in computer science the terminology of file structures has evolved higgledy-piggledy without much concern for consistency, ambiguity, or whether it was possible to make the kind of distinctions that were important.

It was only much later that the need for a well-defined, unambiguous language to describe file structures became apparent. In particular, there arose a need to communicate ideas about file structures without getting bogged down by hardware consideration

Dept. Of ISE,JIT Page| 1 Page| 1

### Introduction to File System

In computing, a file system or file system controls how data is stored and retrieved. Without a file system, information placed in a storage medium would be one large body of data with no way to tell where one piece of information stops and the next begins. By separating the data into pieces and giving each piece a name, the information is easily isolated and identified.

Taking its name from the way paper-based information systems are named, each groups of data is called a “file”. The structure and logic rules used to manage the groups of information and their names is called a “file system”.

There are many different kinds of file systems. Each one has different structure and logic, properties of speed, flexibility, security, size and more. Some file systems have been designed to be used for specific applications. For example, the ISO 9660 file system is designed specifically for optical discs.

File systems can be used on numerous different types of storage devices that use different kinds of media. The most common storage device in use today is a hard disk drive. Other kinds of media that are used include flash memory, magnetic tapes, and optical discs. In some cases, such as with tmpfs, the computer's main memory (random-access memory, RAM) is used to create a temporary file system for short-term use.

Dept. Of ISE,JIT Page| 2 Page| 2

### 1.3 Document processing

Introduction

Documents serve to archive and communicate information. [Document processing](https://www.encyclopedia.com/science-and-technology/computers-and-electrical-engineering/computers-and-computing/document-0) is the activity of operating on information captured in some form of persistent medium. Traditionally, that medium is paper, and documents are bundles of paper with information captured in print or in writing.

[Document processing](https://www.encyclopedia.com/science-and-technology/computers-and-electrical-engineering/computers-and-computing/document-0) may serve to coordinate and conduct business transactions. When a customer submits an order to purchase a certain product, the order becomes a document for processing. The manufacturing company coordinates the activities of acquiring the raw materials, making the product, and finally delivering it to the customer with an invoice to collect payment—all by passing documents from one department to another, from one party to another.

Humans, endowed with the capacity to read, write, and think, are the principal actors in document processing. The invention of the modern [digital computer](https://www.encyclopedia.com/science-and-technology/computers-and-electrical-engineering/computers-and-computing/digital), supported by various key technologies, has revolutionized document processing. Because information can be coded in other media that is read and written by the computer—from **punched cards** in the early 1960s to **magnetic tapes** , disks, and optical CDs (compact discs) today—it is not always necessary for documents to be on paper for processing.

Dept. Of ISE,JIT Page| 3 Page| 1

# CHAPTER 2

# REQUIREMENT ANALYSIS AND DESIGN

**REQUIREMENT ANALYSIS AND DESIGN**

* 1. **Domain Understanding**

The main objective here is to search a given word and know its occurrence along with line number and replace the word if user requests either in aspecif line or in an entire document.

* 1. **Classification of Requirements**

**System Requirements**

* Processors: Intel Atom® processor or Intel® Core™ i3 processor
* Disk space: 100 GB
* Ram: 4 GB
* Operating systems: Windows\* 7 or later, macOS, and Linux
* Python\* versions: 3.6.X or Higher

**Software and Hardware Requirements**

* + - Basic Hardware Requirement:4gb Ram,100gb Rom, dual core processor
    - Programming Languages: Python

Dept of ISE,JIT Page| 4

* 1. **System Analysis**

When the GUI program is executed, it loads a simple Gui which gives 8 different options which are:

Dept. Of ISE,JIT Page| 8

Dept. Of ISE,JIT Page| 5

* Search: This option is used to search all the files in the folder for the given string and returns the occurrences the string along with file number name of the string occurrence.
* Search\_m: This option is used to search all the files in the folder for the given string and returns the occurrences the string along with file number and the line number of the string occurrence.
* Search\_r: This option is used to search all the files in the folder for the given string and returns the occurrences the string along with file number and the line number of the string occurrence. This uses recursive method of search.
* The above three options are case sensitive search options other three options are case insensitive search options.
* Replace all the occurrence of a string in a file: This option searches for the string in the user chosen file and replaces all the occurrences in the whole file with the user wished string.
* Replace the occurrence of a string in a line in a file: This option searches for the string in the user chosen file and replaces the string in the user chosen line in the whole file with the user wished string.

# CHAPTER 3 IMPLEMENTATION

### IMPLEMENTATION

We will be writing a class with 5 functions:

1. Txt\_search(): In this function we will access the folder using os module and we use regular expression module to search the string and return file name and number of files in which the string is found.

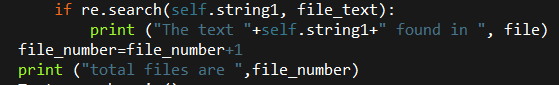


Fig:3.1

To make the function case insensitive we use the following command,

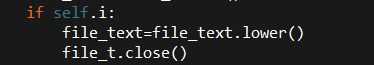


Fig:3.2

1. Txt\_search\_m():In this function we will access the folder using os module and we use regular expression module to search the string and return file name and number of files in which the string is found and this also returns the line numbers if the string found.

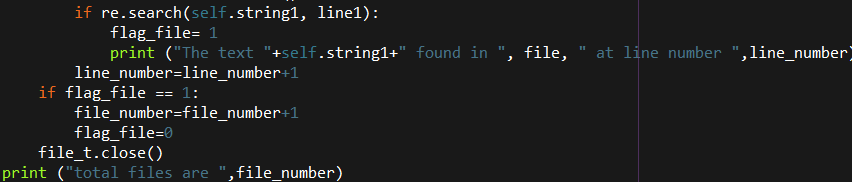


Fig:3.3

The same command is used to make this function case insensitive

1. txt\_search\_r(): In this function we will use simple dialog box of tkinter to take the user input for the searching sting.We initialize a variable to record line number. Then we run a for loop to access every file in the folder then and for each file another for loop to access every line of the file. We then use regular expression package of python to use string search re.search function to search for the string in the line then we increment the line number. We then append the line number and the file name along into a list box which is the GUI used to display the search results

Dept. Of ISE,JIT Page| 6

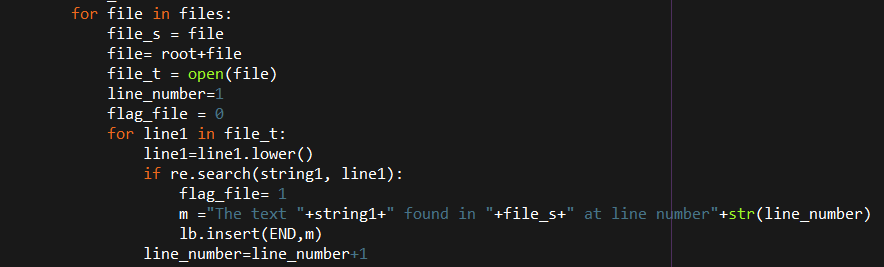


Fig:3.4

The above snapshot is the main logic of the search function

4.replaceall(): In this function we display the list of files and ask the user to choose one of the files from the list. Then we take user input for the string to be replaced and the replace string.Then we access the file and replace all the words occurrence in the file. And when it is done a message success is shown.

Dept. Of ISE,JIT Page| 6 Page| 1

Dept. Of ISE,JIT Page| 7 Page| 1

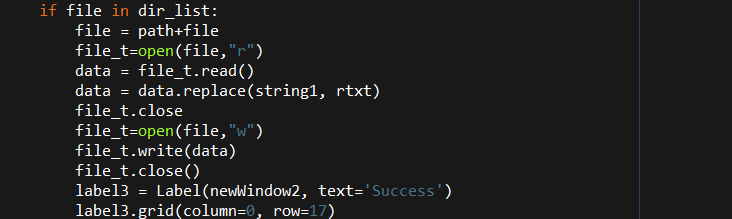


Fig:3.5

The above figure is the snapshot of main logic for the replacing of the string

5.replaceocc(): In this function we display the list of the files like the replaceall() function and take string to be replaced and the replace string from the user. In addition we will also take the line number where the string is to be replaced. Now once the replace button is clicked we run a for loop after accessing the file we read the file line by line. We use a variable to count the lines and we keep increasing it and writing that line to anew file until the user specified line is reached. Then we split the line word by word using strip function and store it as a list. Now we replace the string if present and write it back to a the new file and continue writing all the lines till the end of the file.

Dept. Of ISE,JIT Page| 8 Page| 1

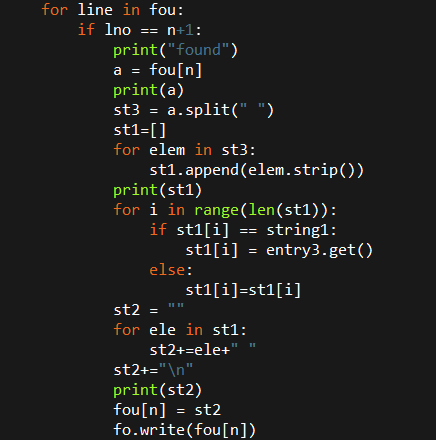
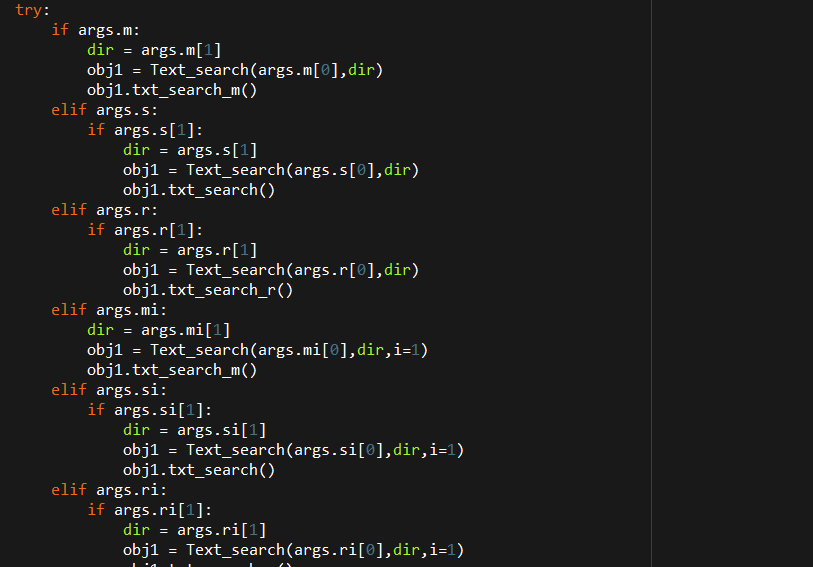


Fig: 3.6

The above figure is the snapshot of the replace at the user specified line.

The main function is written in the executing part of the script. Here we create the main window of the GUI where we place the four buttons that execute the above four functions on clicked.

For the command line interface a separate function is is written as main(). We use argparse module for command line interface.



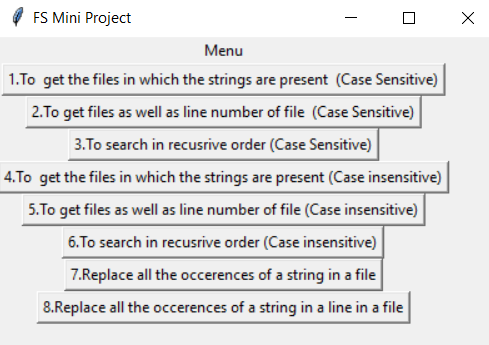
Dept. Of ISE,JIT Page| 9 Page| 1

**CHAPTER 4 RESULT AND ANALYSIS**

**RESULT AND ANALYSIS**

Result and Analysis

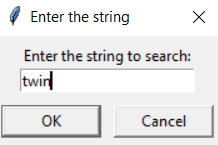
**Figure 5.1:-HOME SCREEN**

****

**Fig: 5.1**

The Above Figure Is the User Interface Home Screen In which it Contains four buttons that perform the four action of the project

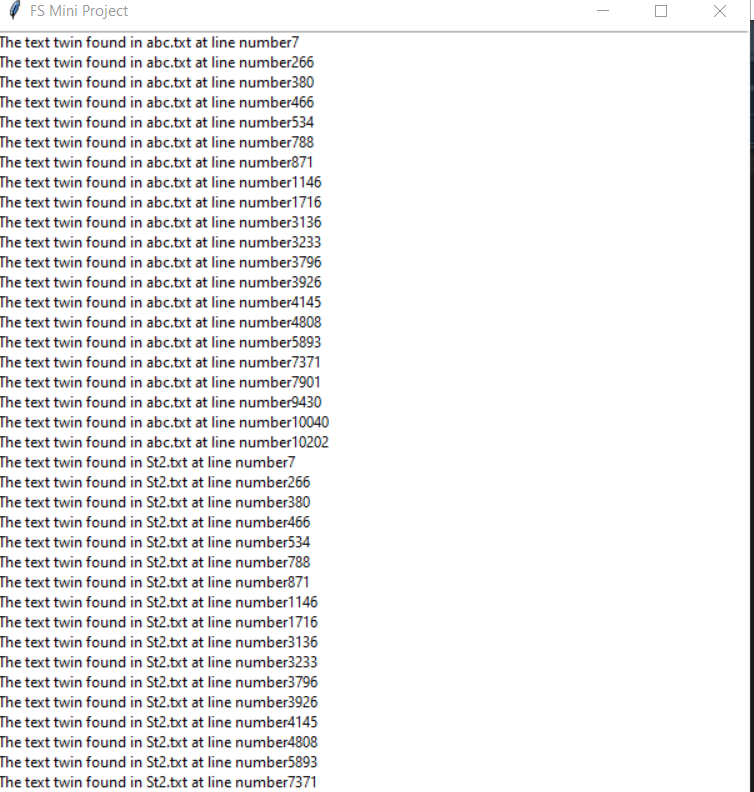
**Figure 5.2 and 5.3:-Search**

****

**Fig 5.2**

This dialog box appears wen search button is clicked. It takes user input for the string to be searched for.

Dept, Of ISE,JIT Page| 10

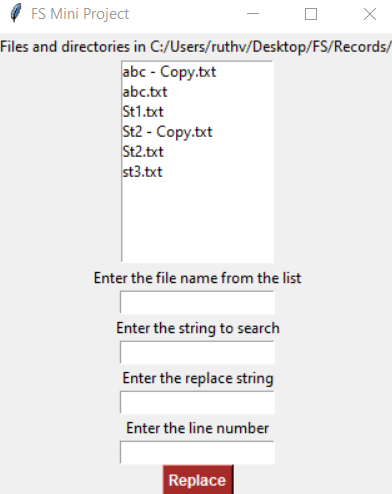
**Fig:5.3**

Dept. Of ISE,JIT Page| 11 Page| 1

This is window appears once the search is completed and this displays the result of the search.

Dept. Of ISE,JIT Page| 12 Page| 1

**Figure 5.4-Replace all the occurrence of a string in a file**

****

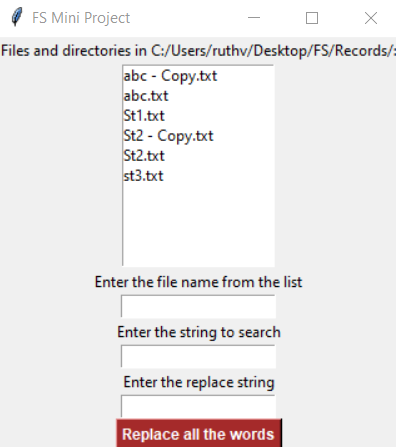
**Fig:5.4**

This window appears on clicking the 3rd button in the window shown in Fig 5.1. It displays the files in folder and takes the user input for file name, string to search and replace string.

Dept, Of ISE,JIT Page| 12

**Figure 5.5:-Replace the occurrence of a string in a line in a file**

Result and Analysis

****

**Fig:5.5**

This window appears on clicking the 4th button in the window shown in Fig 5.1. It displays the files in folder and takes the user input for file name, string to search, replace string and the line number where the string must replaced.

Dept. Of ISE,JIT Page| 13

Figure 5.6: Command promt

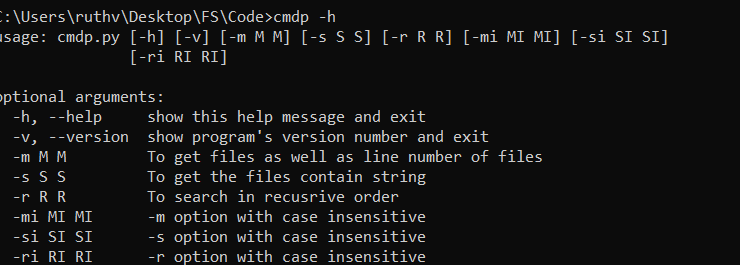


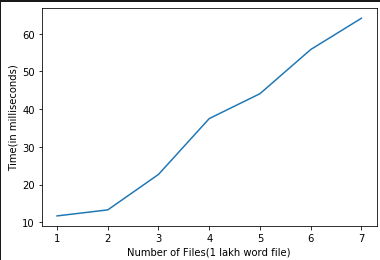
Fig:5.7

The above figure shows the help command which gives details of all the options of the program.

Dept, Of ISE,JIT Page| 14

**Graph**:

Dept, Of ISE,JIT Page| 14



**Fig:5.8**

Result and Analysis

The above graph gives us the time taken versus files of 1 lakh words for the search function to be executed.

**CONCLUSION:**

Dept, Of ISE,JIT Page| 15

Dept, Of ISE,JIT Page| 20

Result and Analysis

In conclusion, Search and replacing of string in a file is a very important part of the Document processing. The files used in this mini-project contain minimum of 1 lakh words. So searching a text and having to replace it is a very huge challenge. This project is aimed at making it simple and less time consuming.

**REFERNCES:**

1. <https://likegeeks.com/python-gui-examples-tkinter-tutorial/>

Dept, Of ISE,JIT Page| 48

1. <https://docs.python.org/3/library/tkinter.html>
2. Tkinter GUI programming by example - David love
3. <https://www.python-course.eu/python_tkinter.php>