NOVEMBER 2022

Menu Based Word search

Assignment 04

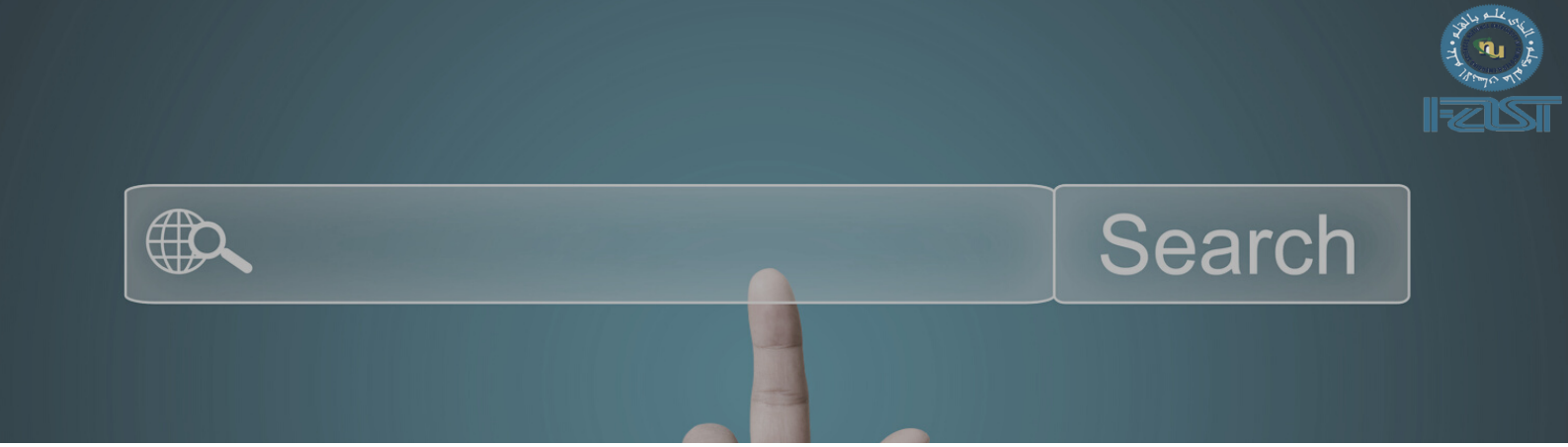
Submitted by:

Najam Ali Abbas (20I-2612)

Jawad ur Rehman(20I-0683)

Submitted to:

Mr.Shoiab Saleem



Language and Tool Used

We have used python for developing this Word search engine because of its easy syntax and extensive libraries. While being incredibly useful for the fields of data science and machine learning, Python is also great for developing graphical user interfaces! In fact, it has many frameworks that even beginners can use to easily get started with developing a GUI. With Python GUI programming we can develop Mobile Applications, Games, and Human-machine Interfaces in Industries. Some Popular GUI frameworks are Tkinter, PySimpleGUI which we have used to create this software, and QT for python.

Tools Used

- **Visual Studio Code**
- **GitHub**
- **Canva**
- **PysimpleGui**
- **Ubantu**

Work Flow of the Project

Python Simple GUI

SimpleGUI is a python component that is being used widely for creating interactive games. Python is a high-level programming language, it is being used for both small and large-scale application development. Python runs on almost all popular operating systems and its interpreters are available for all platforms. With all the advancements in python, SimpleGUI is one of them. Developers all around the world use this component for the development of games and it has gained popularity with the passage of time.

Installing Pysimplegui

Installing Python SimpleGUI on Ubantu

1. To install this tool on Ubantu First you have to install the python se

```
$ sudo apt-get install python-setuptools
```

2. Now Install python-tk pakage otherwise you will get error.

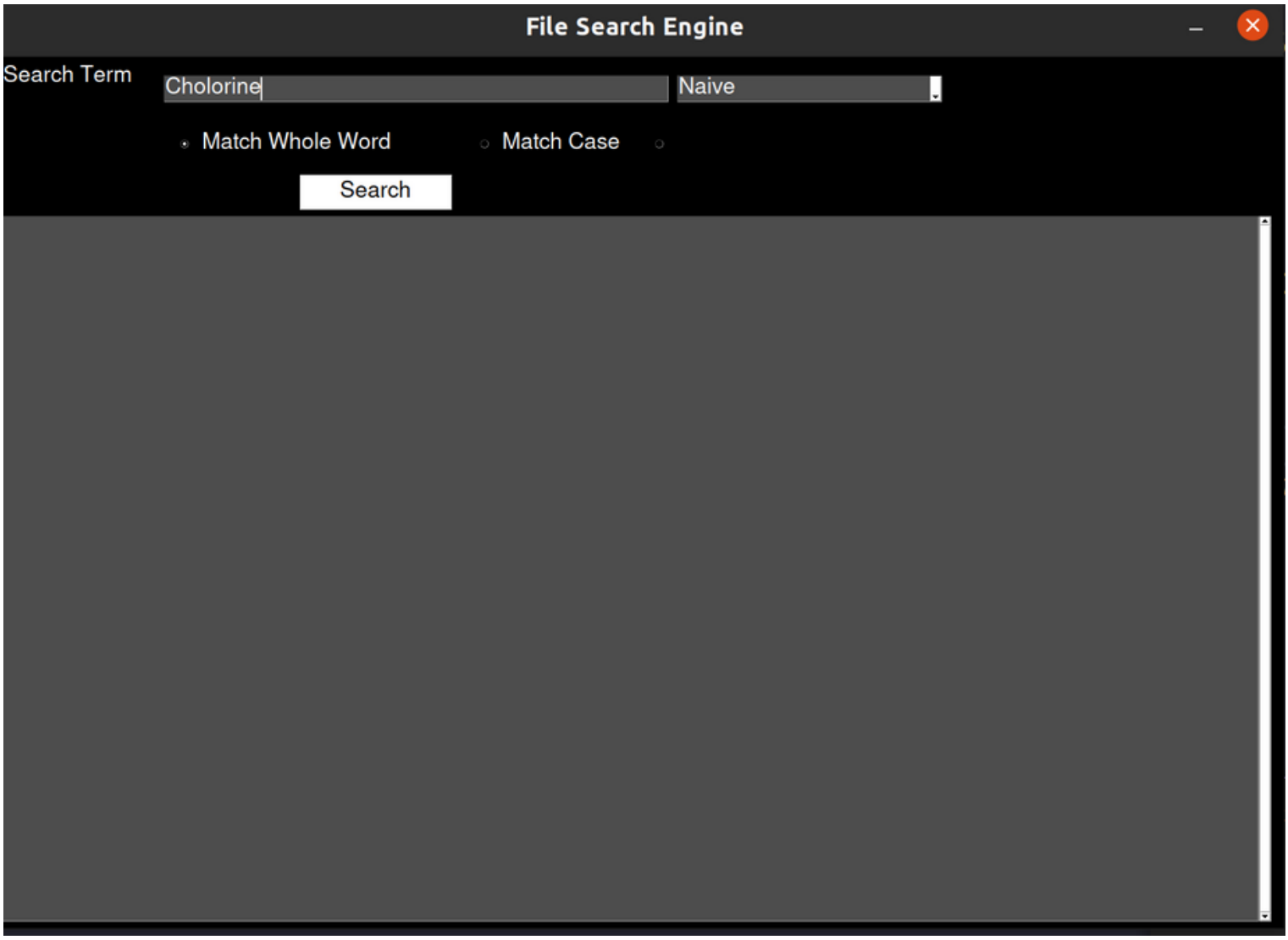
```
$ sudo apt-get install python-tk
```

3. You are almost there , Install simplegui with following command.

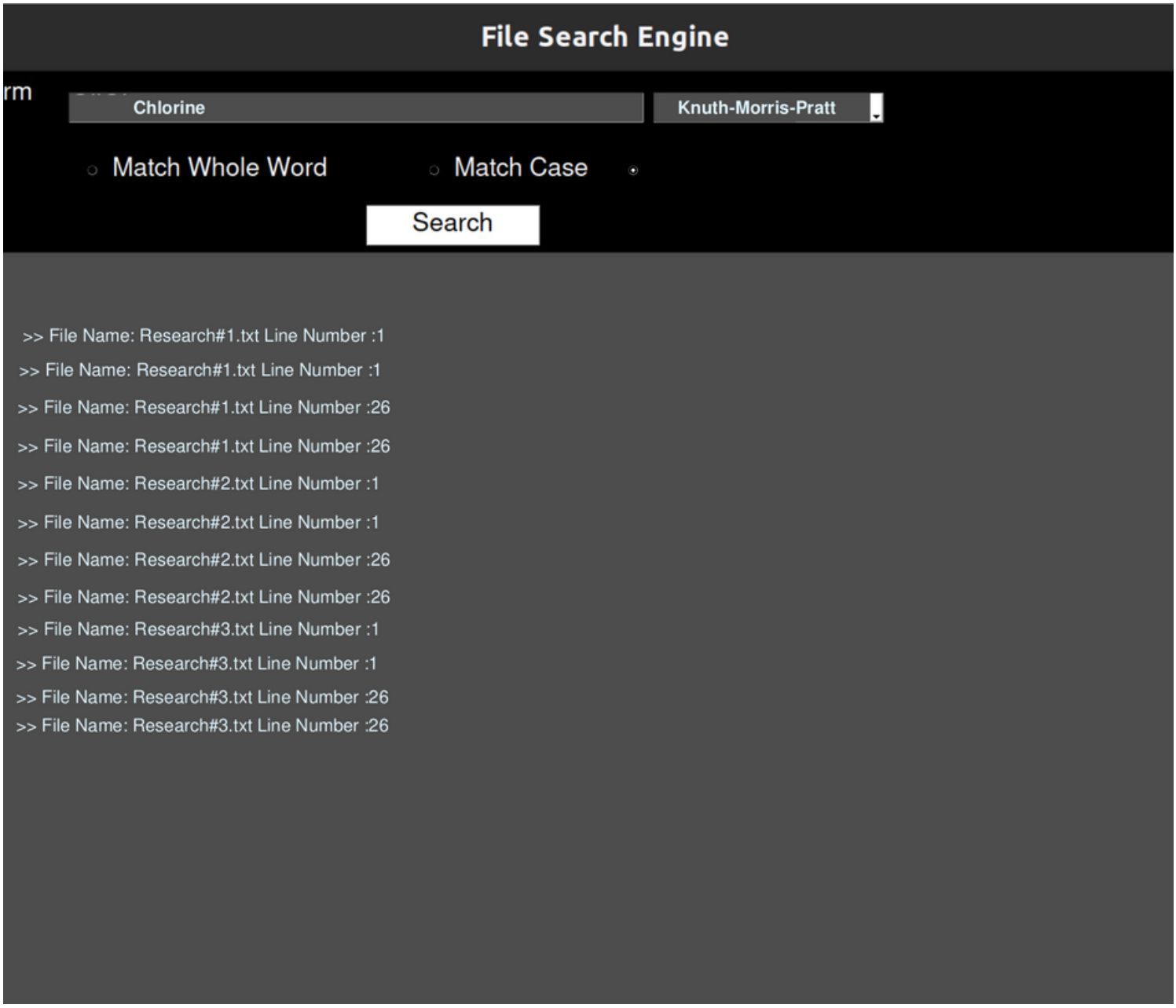
```
$ sudo easy_install simplegui
```

Steps to run the program

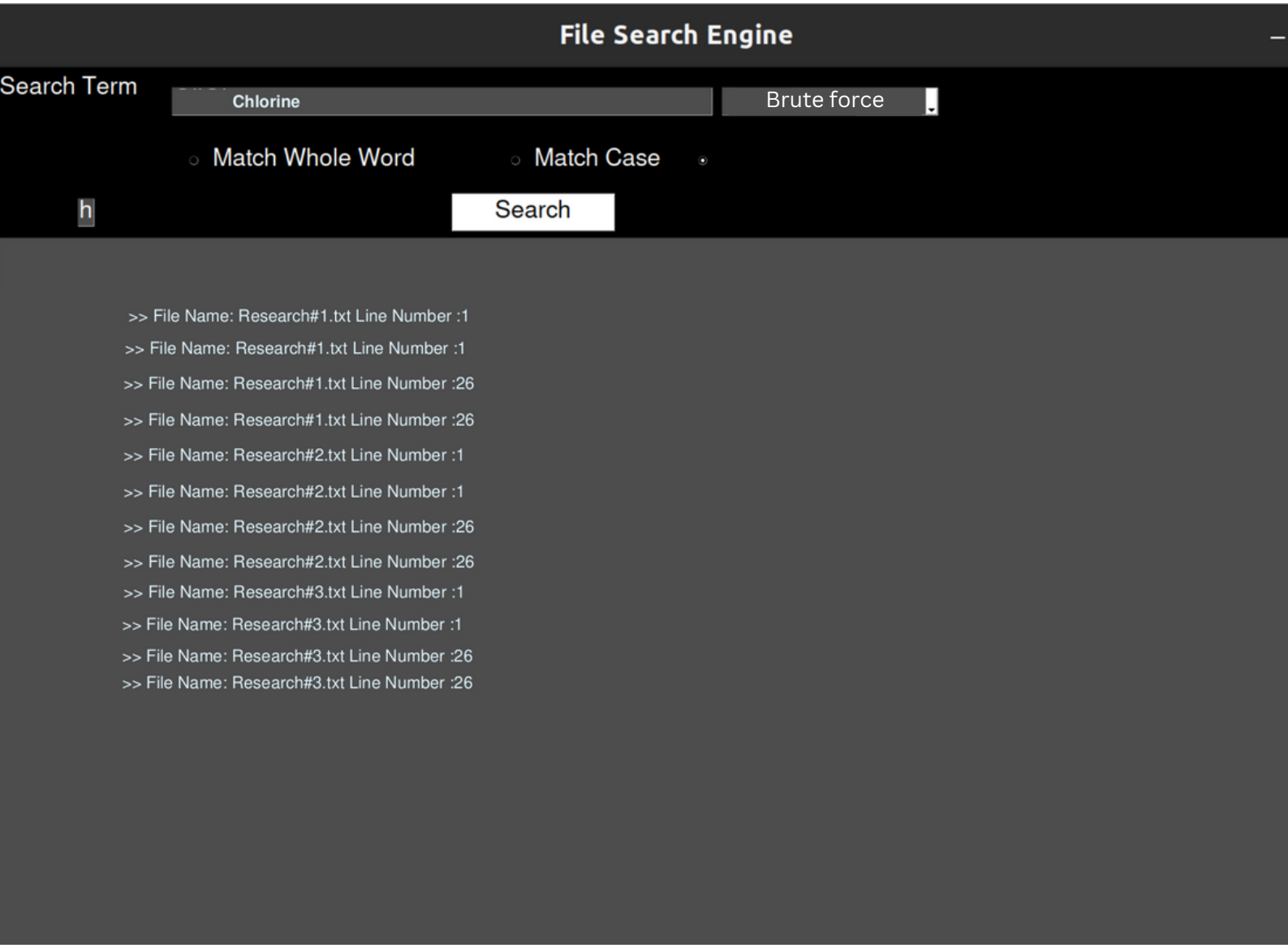
- First of all, Install the PysimpleGUI Library
- Include the Dataset in the project
- Then create another file and include pysimplegui
- Along with the code read data from the file and use string matching algos
- Now Run the project
- You will see this GUI
- Now GUI will pop a message about what you can search



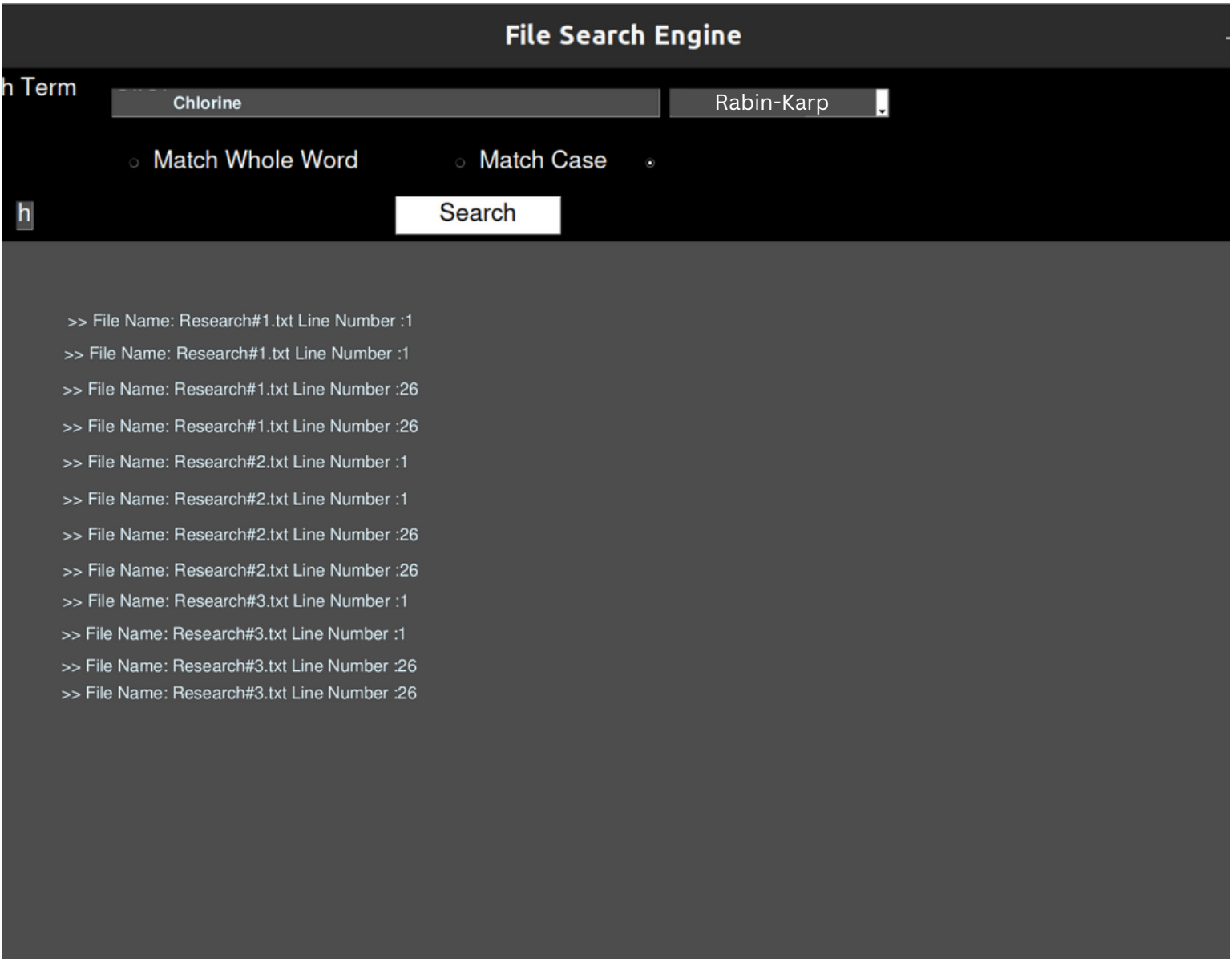
Knuth-Morris-Pratt



Brute-Force


File Search Engine
Search Term
Chlorine
Brute force
Match Whole Word
Match Case
h
Search
>> File Name: Research#1.txt Line Number :1
>> File Name: Research#1.txt Line Number :1
>> File Name: Research#1.txt Line Number :26
>> File Name: Research#1.txt Line Number :26
>> File Name: Research#2.txt Line Number :1
>> File Name: Research#2.txt Line Number :1
>> File Name: Research#2.txt Line Number :26
>> File Name: Research#2.txt Line Number :26
>> File Name: Research#3.txt Line Number :1
>> File Name: Research#3.txt Line Number :1
>> File Name: Research#3.txt Line Number :26
>> File Name: Research#3.txt Line Number :26

Rabin-Karp


File Search Engine
h Term
Chlorine
Rabin-Karp
Match Whole Word
Match Case
h
Search
>> File Name: Research#1.txt Line Number :1
>> File Name: Research#1.txt Line Number :1
>> File Name: Research#1.txt Line Number :26
>> File Name: Research#1.txt Line Number :26
>> File Name: Research#2.txt Line Number :1
>> File Name: Research#2.txt Line Number :1
>> File Name: Research#2.txt Line Number :26
>> File Name: Research#2.txt Line Number :26
>> File Name: Research#3.txt Line Number :1
>> File Name: Research#3.txt Line Number :1
>> File Name: Research#3.txt Line Number :26
>> File Name: Research#3.txt Line Number :26

Write a detailed comparison of the Brute force, RK, and KMP algorithm of String Matching.

Brute force algorithm:

This algorithm uses a simple character-matching algorithm to find the pattern in a given string. The algorithm begins to check characters of the pattern in the string one by one and if any mismatch occurs that word is skipped and subsequently the process begins for the next word.
Worst case is O(m*(n-m+1)).
The best case is O(n).
Some of the advantages of using brute force algorithms are that there's no preprocessing required, no extra space is needed and the comparisons can be done in any order. One of the disadvantages of this algorithm is that the information from shifts is not used again.

Rabin-Karp

In the Rabin Karp algorithm also, we slide the pattern one by one. But here, instead of comparing one by one each character, we compare the hash values of the pattern and the corresponding substring of the string. If the hash value matches, then we print the occurrence.
As the Rabin Karp algorithm is depending on the hash values, we need to calculate the hash value very carefully so that we can minimize the collision, and we store the hash values in a vector so that we don’t have to do the same calculation again and again and, we will get better time complexity.
Worst case O(mn)
Best case O(m + n)

Knuth-Morris-Pratt Algorithm

The KMP matching algorithm uses the degenerating property (pattern having same sub-patterns appearing more than once in the pattern) of the pattern and improves the worst case complexity to O(n). The basic idea behind KMP's algorithm is: whenever we detect a mismatch (after some matches), we already know some of the characters in the text of the next window. We take advantage of this information to avoid matching the characters that we know will anyway match.

Worst case O(m+n)
Best case O(m)

Links and Refrences

- **https://www.seeedstudio.com/blog/2021/07/19/what-is-python-gui-programming-uses-frameworks-**
- **https://www.geeksforgeeks.org/**
- **https://readthedocs.org/projects/pysimplegui/**
- **Lectures slides**


THE
END