



**Elaborado por: Ricardo Carrillo Sanchez**

## **Descripción de los mecanismos de sincronización empleados**

Para la solución de este problema se empleó un semáforo que tiene la función de indicar cuando un hilo concreto su proceso de búsqueda, pudiendo ser este positivo o negativo, una barrera que tiene la función de no permitir el paso a hilos de manera simultánea a la sección crítica que comprende a algunas variables de estado y el archivo en el que se escribirán las coincidencias encontradas

### **Lógica de operación:**

- **Identificación del estado compartido (variables o estructuras globales)**

El estado compartido consta principalmente de un archivo de texto plano en el cual cada uno de los hilos (pregunta) puede almacenar las coincidencias que haya encontrado con la pregunta que se le asignó. Otras variables también conforman dicho estado, pero estas únicamente sirven para brindarle información al controlador para poder realizar la administración de estos hilos.

- **Descripción algorítmica del avance de cada hilo/proceso y descripción de la interacción entre ellos (sea mediante los mecanismos de sincronización o de alguna otra manera)**

Hilos Pregunta: una vez creados, estos comienzan su labor de búsqueda de coincidencias con el “texto brindado”, una vez encontradas, este accede a la región crítica donde por medio de la barrera se le brindara el “permiso” por parte del controlador de sobrescribir el contenido del archivo de respuestas. En caso de no encontrar coincidencias, el hilo accede a la región crítica para hacerle llegar al controlador que no encontró coincidencias y que este acabara con su labor. Una vez terminados los casos anteriores se le notifica al controlador.

Cabe destacar que la región crítica está protegida por una barrera y un mutex.

Controlador: una vez creado mantiene comunicación con los hilos pregunta, se les brinda acceso a la región crítica. Este sin embargo, se queda esperando por medio de un semáforo a que un hilo pregunta termine, sin importar si este encontró coincidencias o no. Una vez terminados todos los hilos, este sin más imprime un resumen de los hilos que encontraron una coincidencia con el texto y de los que no.

## Descripción del entorno de desarrollo, suficiente para reproducir una ejecución exitosa

- ¿Qué lenguaje emplean? ¿Qué versión?

Para el desarrollo de este simple sistema se utilizó el lenguaje de programación Python en su versión 3.9 y una versión de pip 20.3.4

```
iroh@iroh:~$ pip3 --version
pip 20.3.4 from /usr/lib/python3/dist-packages/pip (python 3.9)
iroh@iroh:~$ python3 --version
Python 3.9.2
iroh@iroh:~$ |
```

- ¿Qué bibliotecas más allá de las estándar del lenguaje?

Se utilizaron dos bibliotecas para su ejecución:

1) lorem\_text:

Para instalar lorem\_text lo puede hacer ejecutando el siguiente comando en la terminal de linux:

```
pip install lorem-text
```

Para más información: [lorem-text · PyPI](#)

2) numpy

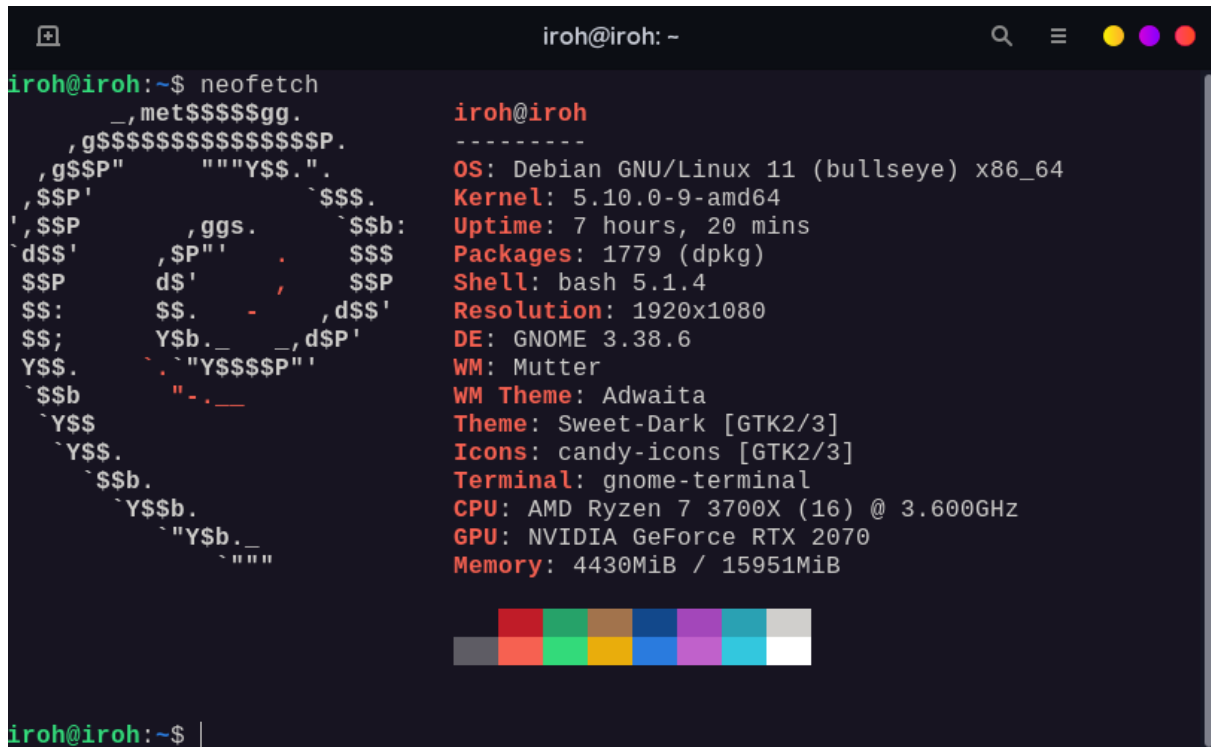
Para instalar numpy lo puede hacer ejecutando el siguiente comando en la terminal de linux:

```
pip install numpy
```

Para más información: [Installing NumPy](#)

- ¿Bajo qué sistema operativo / distribución lo desarrollaron y/o probaron?

El programa se desarrolló y probó en una computadora con las siguientes características:



```
iroh@iroh:~$ neofetch
_,met$$$$$gg.
,g$$$$$$$$$$$$$P.
,g$$$P" ""Y$$.
,$$P' `$$$
',$$P ,ggs. `$$b:
`d$$' ,,$P"' $$$
$$$ d$' , $$$
$$: $$ - ,d$$'
$$; Y$b._ _ ,d$P'
Y$$ "Y$$$$P"
`$$b "-. _
`Y$$
`Y$$.
`$$b.
`Y$$.
`"Y$b. _
`""

iroh@iroh
-----
OS: Debian GNU/Linux 11 (bullseye) x86_64
Kernel: 5.10.0-9-amd64
Uptime: 7 hours, 20 mins
Packages: 1779 (dpkg)
Shell: bash 5.1.4
Resolution: 1920x1080
DE: GNOME 3.38.6
WM: Mutter
WM Theme: Adwaita
Theme: Sweet-Dark [GTK2/3]
Icons: candy-icons [GTK2/3]
Terminal: gnome-terminal
CPU: AMD Ryzen 7 3700X (16) @ 3.600GHz
GPU: NVIDIA GeForce RTX 2070
Memory: 4430MiB / 15951MiB

iroh@iroh:~$
```

El entorno que se empleó para su desarrollo fue el editor de código visual studio code con sus siguientes características:



```
Visual Studio Code
Visual Studio Code

Version: 1.61.2
Commit: 6cba118ac49a1b88332f312a8f67186f7f3c1643
Date: 2021-10-19T14:58:13.605Z (1 wk ago)
Electron: 13.5.1
Chrome: 91.0.4472.164
Node.js: 14.16.0
V8: 9.1.269.39-electron.0
OS: Linux x64 5.10.0-9-amd64
```

## Pruebas de ejecución:

```
Soy (1) -> buscando...
Soy (1) -> encuentre 1 coincidencias
Soy (1) -> encuentre 2 coincidencias
Soy (1) -> agregare algo a respuestas.txt
CONTROLADOR -> tengo 9 preguntas pendientes
Soy (7) -> buscando...
Soy (7) -> encuentre 1 coincidencias
Soy (7) -> encuentre 2 coincidencias
Soy (7) -> encuentre 3 coincidencias
Soy (7) -> agregare algo a respuestas.txt
Soy (8) -> buscando...
Soy (8) -> no encuentre coincidencias para la pregunta 8
Soy (9) -> buscando...
CONTROLADOR -> tengo 7 preguntas pendientes
Soy (9) -> encuentre 1 coincidencias
CONTROLADOR -> tengo 7 preguntas pendientes
Soy (9) -> encuentre 2 coincidencias
Soy (9) -> encuentre 3 coincidencias
Soy (9) -> encuentre 4 coincidencias
Soy (9) -> encuentre 5 coincidencias
Soy (9) -> agregare algo a respuestas.txt
CONTROLADOR -> tengo 6 preguntas pendientes
Soy (2) -> buscando...
Soy (2) -> encuentre 1 coincidencias
Soy (2) -> encuentre 2 coincidencias
Soy (4) -> buscando...
Soy (2) -> agregare algo a respuestas.txt
Soy (4) -> encuentre 1 coincidencias
Soy (5) -> buscando...
Soy (6) -> buscando...
Soy (5) -> no encuentre coincidencias para la pregunta 5
Soy (6) -> encuentre 1 coincidencias
Soy (4) -> agregare algo a respuestas.txt
Soy (3) -> buscando...
Soy (10) -> buscando...
Soy (6) -> encuentre 2 coincidencias
CONTROLADOR -> tengo 3 preguntas pendientes
Soy (3) -> encuentre 1 coincidencias
Soy (10) -> no encuentre coincidencias para la pregunta 10
Soy (6) -> encuentre 3 coincidencias
CONTROLADOR -> tengo 3 preguntas pendientes
Soy (6) -> encuentre 4 coincidencias
CONTROLADOR -> tengo 2 preguntas pendientes
```

```
Preguntas encontradas: [1, 7, 9, 2, 4, 3, 6]
```

```
Preguntas no encontradas: [8, 5, 10]
```

```
iroh@iroh:~/Documents/sistop$ |
```