UNIVERSIDAD AUTONOMA DEL ESTADO DE MÉXICO

FACULTAD DE INGENIERIA

Sistemas Operativos:

Proyecto 1: Simulador de Calendarización

Alumno:

Victor Manuel Jiménez Rosas

**Introducción**

Para la realización del proyecto se utilizó el sistema operativo Windows y el lenguaje de programación Python debido a la facilidad para trabajar con archivos y a la sencillez que provee el lenguaje tanto en instalación como en ejecución y codificación.

**Instalación de Python**

Para la instalación de Python es necesario descargar el archivo desde <https://www.python.org/> y luego seguir la instalación recomendada por la aplicación.

NOTA IMPORTANTE: INDISPENSABLE INSTALAR PYTHON 3

**Configuración**

El archivo de configuración llamado “calendarizador.conf” tiene la siguiente estructura:

// # de Procesos

numero\_procesos **{numero}**

// media

promedio **{numero}**

// desviacion estándar

desv\_est **{numero}**

// procesos # Bloqueo E/S #Bloqueo Quanto

proceso **{numero}** **{numero}** **{prioridad}**

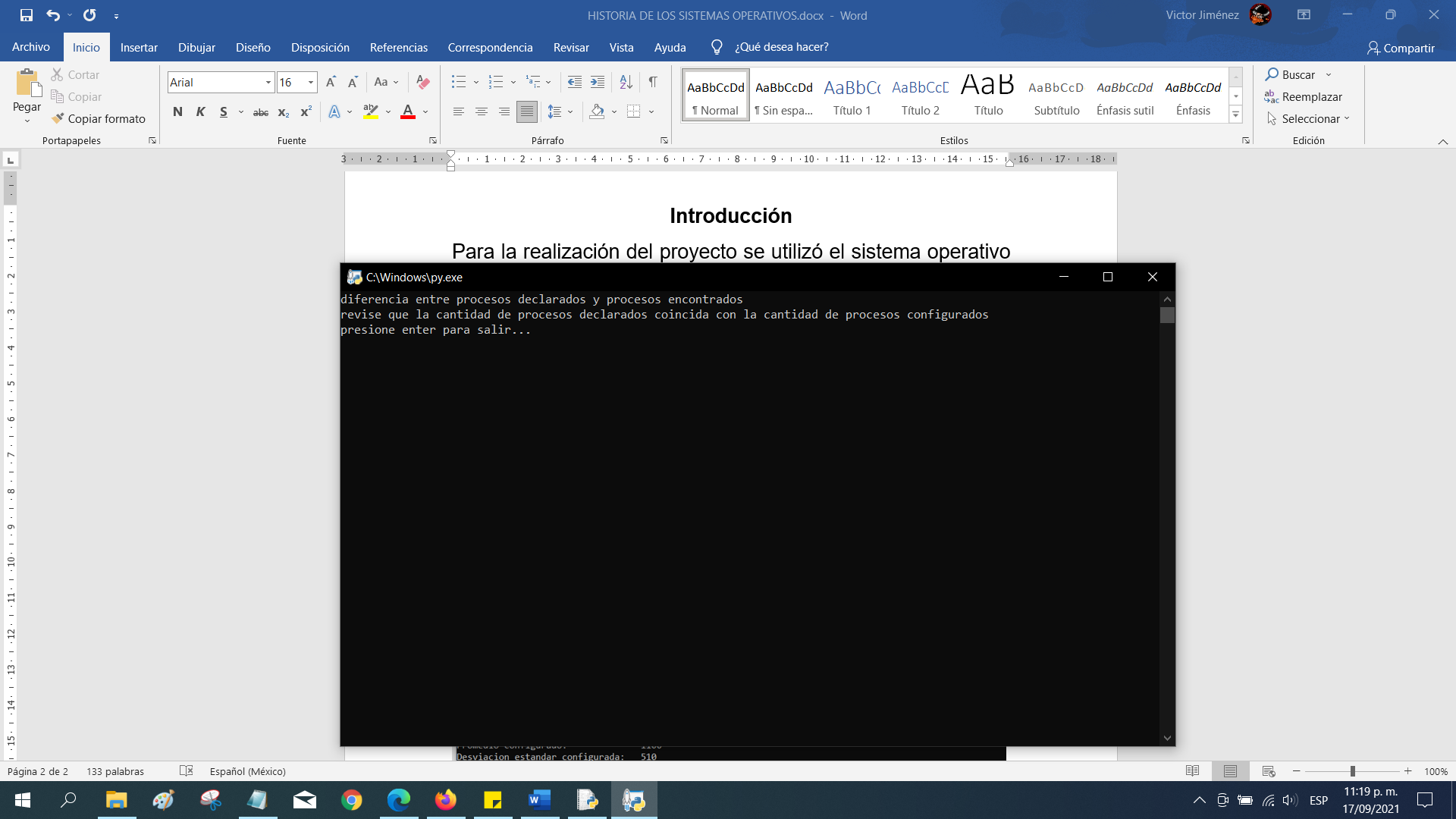
**{…}**

// duracion de la simulacion en milisegundos

tiempo\_ejec **{numero}**

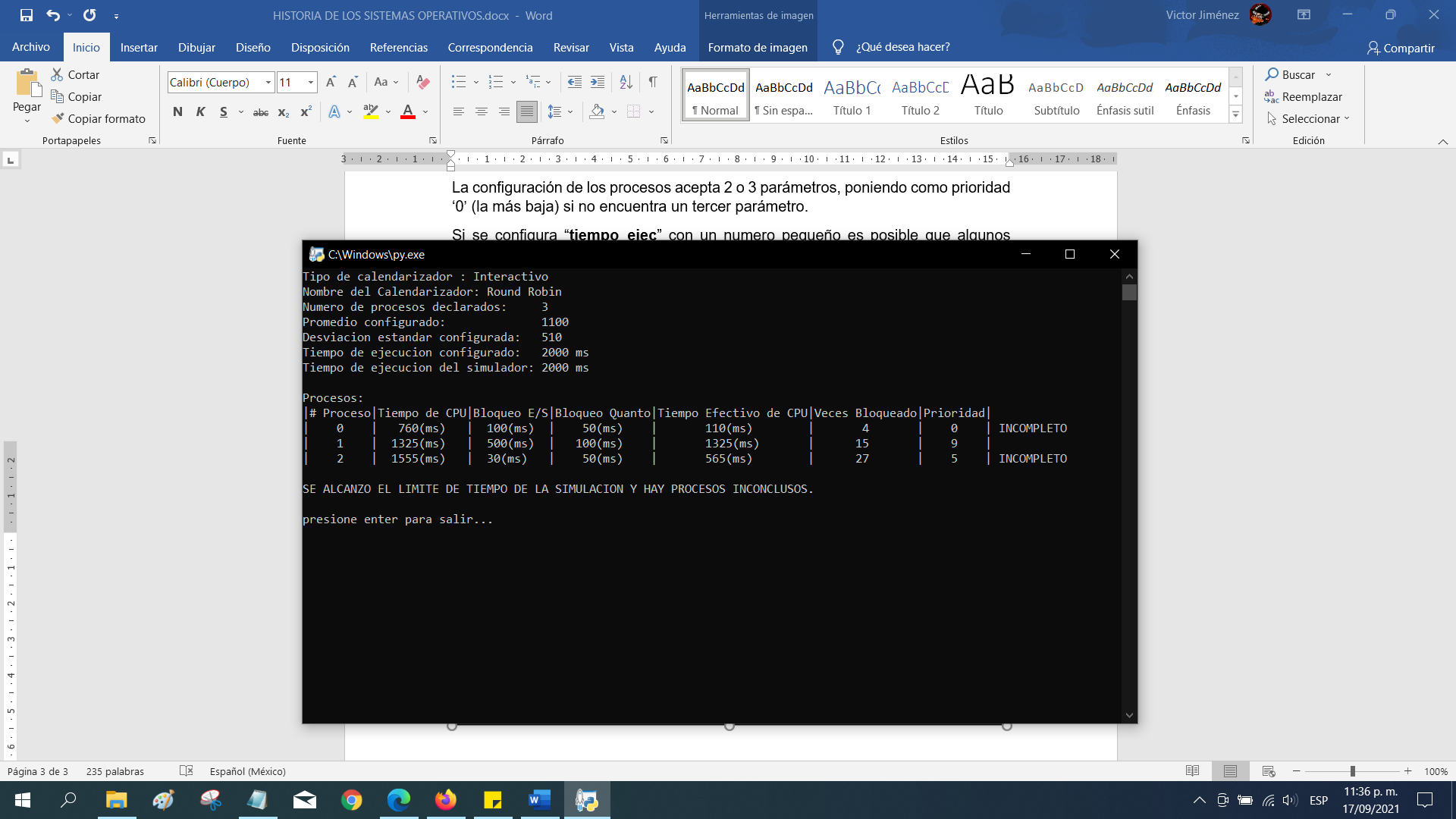
Siendo **{numero}** un numero entero cualquiera, **{prioridad}** un numero entero del 0 al 10 y **{…}** una repetición de la estructura “proceso”.

Es muy necesario que en el archivo de configuración se declare el mismo número de procesos en “**numero\_procesos**” que los procesos configurados con la palabra reservada “**proceso**” pues de otro modo el programa mostrara un mensaje alertando de la mala configuración introducida y terminara su ejecución.



La configuración de los procesos acepta 2 o 3 parámetros, poniendo como prioridad ‘0’ (la más baja) si no encuentra un tercer parámetro.

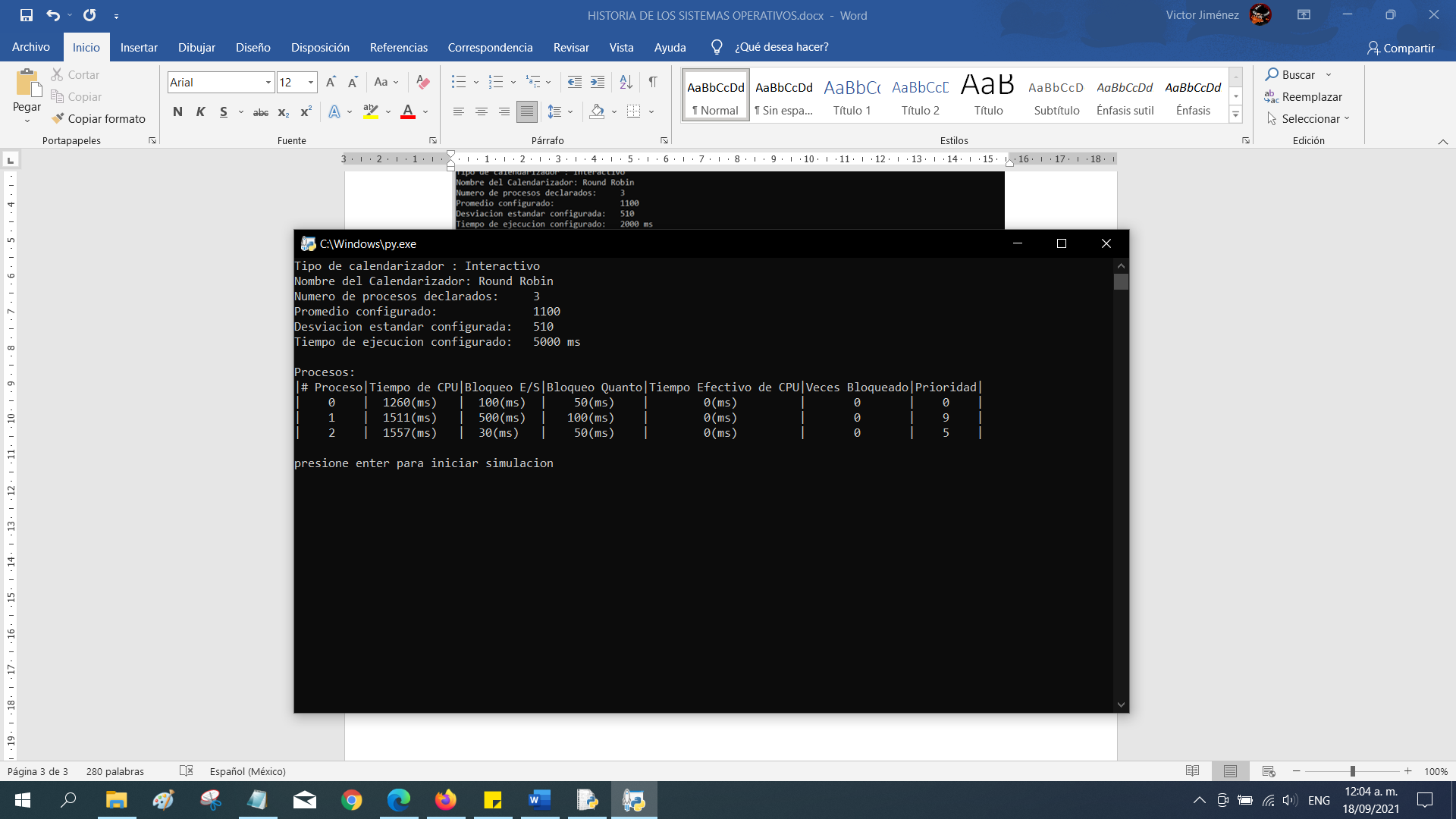
Si se configura “**tiempo\_ejec**” con un numero pequeño es posible que algunos procesos no finalicen, mostrando a un lado del proceso en el archivo “**ResumenResultados.txt**” y en la pantalla una leyenda con la palabra “**INCOMPLETO**”.



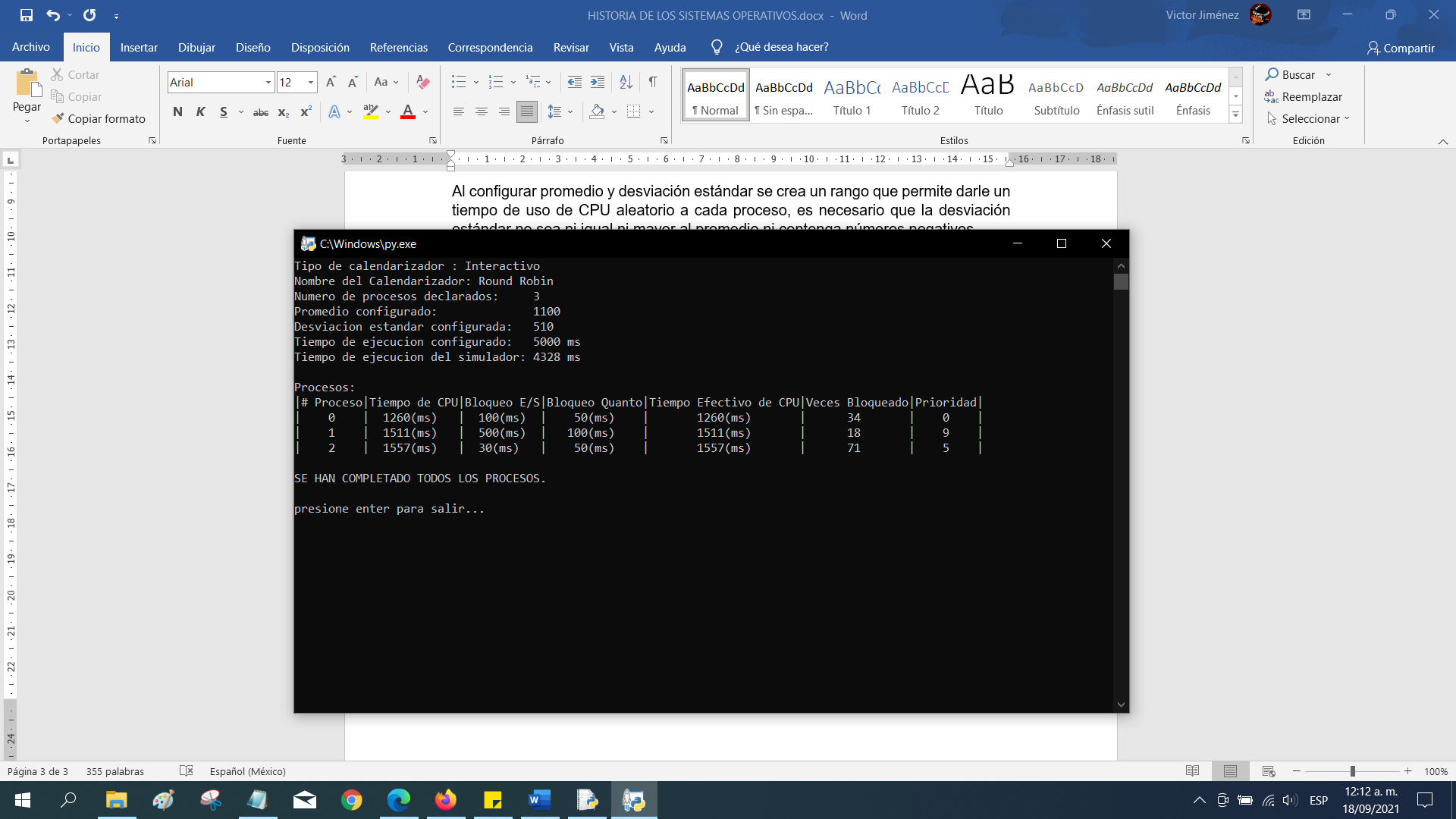
Al configurar promedio y desviación estándar se crea un rango que permite darle un tiempo de uso de CPU aleatorio a cada proceso, es necesario que la desviación estándar no sea ni igual ni mayor al promedio ni contenga números negativos.

Ejecucion

Para ejecutar el programa solo hay que hacer doble clic en el archivo llamado “**programa.py**” que mostrara un resumen de la configuración y la configuración inicial de los procesos ingresados.



Al pulsar la tecla “Enter” el programa iniciará y comenzará la simulación, al término de esta simulación se mostrará nuevamente el resumen de la configuración y los procesos, pero esta vez con el tiempo efectivo y el número de veces que se ha bloqueado.



Durante la ejecución se guarda cada acción del simulador en un archivo llamado “**ResumenProcesos.txt**” independientemente de si se completa o no el proceso o si se cierra la aplicación.

Al finalizar la ejecución del simulador se crea un archivo llamado “**ResumenResultados.txt**” con un resumen muy similar al mostrado en pantalla.