

TAREA NO. 4

Profesor: Audie Ruiz Ortega

COLEGIO UNIVERSITARIO DE CARTAGO COLEGIO UNIVERSITARIO DE CARTAGO

Tabla de contenido

1.1	La tarea es individual	1
1.2	Sistema Operativo: Linux.....	1
1.3	Lenguaje de programación: C.....	1
1.4	La documentación solicitada debe incluir como mínimo:	1
1.5	Código fuente del programa	1
1.6	Rastreo de las llamadas al sistema	1
1.7	Respectivo reporte de llamadas en archivos texto.....	1
2	Condiciones.....	1
2.1	La tarea es individual (se revisarán procesos)	1
2.2	Sistema Operativo: Linux.....	1
2.3	Rastreos de llamadas con STRACE.....	1
2.4	Valor de la tarea 30% de la nota final del curso	1
2.5	Fecha de entrega: 20/11/2023 grupo lunes, 22/11/2023 grupo miércoles antes de la 6 pm al correo institucional del profesor	1
3	PARTE I-Bases de la tarea.....	1
4	PARTE II-Generaciones de salidas.....	1
5	PARTE III-Desarrollo.....	2
6	PARTE IV-Indicaciones finales.....	3

1 Aspectos generales

- 1.1 La tarea es individual
- 1.2 Sistema Operativo: Linux (WSL)
- 1.3 Lenguaje de programación: C
- 1.4 La documentación solicitada debe incluir:
 - 1.4.1 Código fuente del programa
 - 1.4.2 Rastreo de las llamadas al sistema
 - 1.4.3 Respectivo reporte de llamadas en archivos texto
 - 1.4.4 Alguna otra especificación

2 Condiciones

- 2.1 La tarea es individual (se revisarán procesos)
- 2.2 Sistema Operativo: Linux
- 2.3 Rastreos de llamadas con STRACE
- 2.4 Valor de la tarea 30% de la nota final del curso
- 2.5 Fecha de entrega: 19/08/2024 grupo lunes, 21/08/2024 grupo miércoles antes de la 6 pm al correo institucional del profesor

3 PARTE I-Bases de la tarea

Para esta tarea, usted debe basarse en la tarea no. 3, la cual, debe estar optimizada y corriendo sin errores. **Valor 10 pts.**

Todas las llamadas que a continuación se solicitan, deben guardarse en archivos texto, según el nombre indicado en cada punto.

4 PARTE II-Generaciones de salidas

En esta parte, debe ejecutar su programa y, debe generar las siguientes llamadas al sistema, y guardarlas en archivos tipo texto según indica cada punto.

- 1. Las marcas de tiempo se muestran al comienzo de cada línea de salida. **Archivo tiempo.txt. Valor 5 pts.**

2. Las duraciones de tiempo de cada línea de llamada del sistema. **Archivo duratiempo.txt. Valor 5 pts.**
3. La hora a la que se llamó cada llamada del sistema, con una resolución de microsegundos. **Archivo hora.txt. Valor 5 pts.**
4. Debe identificar el ID de cada proceso (de existir varios) durante la ejecución de su programa. **Archivo Identificador.txt. Valor 5 pts.**
5. Debe generar un informe sobre las llamadas al sistema generadas, durante la ejecución de su programa. **Archivo Informe.txt. Valor 5 pts.**
6. Imprimir puntero de instrucción durante la llamada al sistema. **Archivo Puntero.txt. Valor 5 pts.**
7. Mostrar hora del día para cada línea de salida de traza. **Archivo Horalinea.txt. Valor 5 pts.**
8. Rastrear todas las llamadas al sistema que implican asignación de memoria. **Archivo memoria.txt. Valor 10 pts.**
9. Desplegar la información de depuración. **Archivo depura.txt. Valor 10 pts.**
10. Despliegue la ayuda del comando **strace**. **Archivo ayuda.txt. Valor 5 pts.**

5 PARTE III-Desarrollo

Para el desarrollo de la Parte II, debe generar un programa en C desde la terminal Linux, que despliegue un menú de opciones para cada uno de los puntos.

EJEMPLO:

MENU DE OPCIONES

1. Marcas de Líneas
2. Tiempos
3. Horas y micro segundos
4. Identificador de procesos
5. Informes
6. Punteros
7. ...
8.
9.

10. Terminar

Etc.

Para cada ejecución de las opciones individualmente, debe implementar lo siguiente:

1. El programa de la tarea no. 3, debe ejecutarse con un hilo, reservas de memoria, bloqueos y desbloques. **Valor 5 pts.**
2. Debe liberar todo recurso, una vez finalizada la ejecución de cada opción y del programa (tarea no. 3). **Valor 5 pts.**

6 PARTE IV-Indicaciones finales

Debe recordar, que, para cada una de las opciones, debe ejecutar el programa de la tarea no. 3. Si lo desea, ejecutará el programa una sola vez al principio. En estos casos, el estudiante puede, implementar mejoras a la ejecución, es decir, decide si ejecuta el programa una vez, o cada vez que se escoja una opción.

Además, debe tener la opción de finalizar el programa y, cada vez que termina la ejecución de alguna de las opciones, debe desplegar su resultado (mostrar el archivo texto correspondiente). Una vez desplegado el archivo texto, debe mostrar opción para continuar (volver al menú, o finalizar), ejemplo, **Desear continuar (S/N)**, si decide S, volverá al menú principal, caso contrario, termina la ejecución del programa.

Para las revisiones de las tareas, se usará la máquina virtual **WSL en Ubuntu Linux**, por lo que, para asegurar su funcionalidad se recomienda usar el **WSL**. Si decide utilizar otras herramientas, debe indicarlo, como, por ejemplo, si usa otra distribución de Linux, u otra máquina virtual. Si utiliza otra distribución o máquina virtual, se recomienda llevar su equipo para minimizar cualquier falla funcional.

No debe olvidar que, la versión de su tarea a usar (si decide emplear su equipo personal), debe ser la misma enviada al correo del profesor, caso contrario, se ejecutara la remitida al profesor.

Por último, si se detectan programas idénticos en estructura, diseño, código, u otro aspecto, se tomará y analizará mediante las herramientas indicadas, para determinar o descartar plagios. De detectarse plagio o copia, se aplicará lo indicado en el reglamento de la institución.

La fecha de entrega, es solamente la indicada, no se recibirán tareas en fechas posteriores excepto en casos de situaciones especiales. En caso de estas situaciones especiales, debe presentar los documentos probatorios según el reglamento interno de la institución.