

Prueba técnica aspirante a desarrollador Junior de Backend, LAP

Prerrequisitos

La siguiente prueba debe ser desarrollada teniendo en cuenta (de manera OBLIGATORIA) los siguientes aspectos:

- Debe ser desarrollada en .NET Core 5, con C# 9, sin importar el IDE que utilice
- El código fuente, resultado de esta prueba, debe ser entregado ANTES de 60 horas, luego de recibir esta prueba en su correo.
- Los resultados de la prueba deben ser subidos a GitHub, en un repositorio PUBLICO, a una cuenta personal del aspirante (si no posee cuenta, créela). El link donde ha quedado subido el código de la prueba, debe ser enviado al siguiente correo: jfmachado@equitel.com.co. Ese correo debe ser enviado ANTES de la culminación del plazo (puede ser un minuto antes, no importa). Debe quedar claro que si el aspirante no envía el correo con la localización del código de la prueba en Github para el momento en el que el plazo de entrega se cumpla, el aspirante quedará inmediatamente descalificado.
- Todo el código en C# que escriba el aspirante para la solución de esta prueba debe estar debidamente ordenado y documentado.
- Puede usar código de Internet, no pasa nada siempre y cuando sea capaz de sustentar su uso y funcionamiento AL DETALLE a la hora de hacer la sustentación.
- La aplicación a desarrollar debe ser una aplicación de CONSOLA que corra de manera igual en Linux y en Windows.
- Tenga en cuenta que en esta prueba se calificará no sólo la correcta funcionalidad del programa final, si no también la limpieza y claridad del código fuente, la aplicación adecuada de patrones de diseño, y obviamente, el buen uso del lenguaje C# 9, del framework (.Net Core 5) y de las clases y librerías de 3eros que decida usar para solucionar el problema.
- Lo que más peso tendrá en la calificación final será el PERFORMANCE de su aplicación y el buen y reducido uso de la memoria RAM del computador, por lo tanto utilice todo el arsenal que tiene a su disposición en el framework (.Net Core 5) y en el lenguaje (C# 9) para cumplir lo mejor que pueda con este objetivo.

Descripción del problema

Desarrolle un programa que lleve al cabo las siguientes tareas EN PARALELO usando diferentes threads de ejecución (ojo, es indispensable que las acciones se ejecuten en paralelo y cada una en su propio thread (es decir, las tareas 1, 2, 3, etc deben arrancar a ejecutarse al mismo tiempo, y deben ejecutarse de manera simultánea), no pueden ser ejecutadas de manera secuencial (por ejemplo, primero la #1, una vez termine, la # 2, etc)):

1-) Leer un archivo de texto, que se encuentre localizado en el equipo y contar las palabras de dicho archivo que terminen con la letra n. Escribir el número de palabras encontradas en OTRO archivo de texto, e imprimir dicho número en pantalla.

2-) Leer EL MISMO archivo de texto de la acción 1 (ojo, el mismo archivo físico, no una copia exacta del archivo) y esta vez, contar el número de oraciones que contengan más de 15 palabras. Entiéndase por oración aquel conjunto de palabras que terminan en un punto SEGUIDO. Escribir el número de oraciones de más de 15 palabras encontradas en el MISMO archivo de texto donde se escribió (o escribirá) el resultado de la tarea 1, e imprimir dicho número en pantalla.

3-) Leer EL MISMO archivo de texto de la acción 1 (ojo, el mismo archivo físico, no una copia exacta del archivo) y esta vez, contar el número de párrafos que contenga el archivo. Entiéndase por párrafo aquel conjunto de palabras que terminan en un punto APARTE. Escribir el número de párrafos encontrados en el MISMO archivo de texto donde se escribió (o escribirá) el resultado de la tarea 1, e imprimir dicho número en pantalla.

4-) Leer EL MISMO archivo de texto de la acción 1 (ojo, el mismo archivo físico, no una copia exacta del archivo) y esta vez, contar el número de caracteres alfanuméricos distintos a n o N que contenga el archivo. Entiéndase por caracteres alfanuméricos toda letra que pertenezca a los rangos [A Z], [a z] y [0 9]. Es decir ni el punto ni la coma, ni el espacio, etc son considerados caracteres alfanuméricos. Escribir el número de alfanuméricos distintos a n o N encontrados en el MISMO archivo de texto donde se escribió (o escribirá) el resultado de la tarea 1, e imprimir dicho número en pantalla.

NOTA1: Dentro de CADA thread debe llevarse a cabo la TOTALIDAD de la tarea (es decir, desde leer el archivo de texto, hasta escribir el resultado en pantalla y en el archivo de salida).

NOTA2: El orden en el que los resultados se imprimen en pantalla o se escriben en el archivo texto de salida no es relevante, es decir, ya que las tareas inician al mismo tiempo, no se puede predecir cuál de ellas terminará antes que las otras, así que podría darse el caso que la tarea 3 terminara antes que la tarea 1 y por lo tanto el resultado de la tarea 3 se imprima en pantalla y se escribirá en el archivo texto antes que el resultado de la tarea 1. Lo importante es que los 4 resultados se impriman en pantalla y se escriban en el MISMO archivo texto de salida (en cualquier orden y bien identificados para saber a qué tarea pertenece cada resultado).

NOTA3: El texto contenido en el (único) archivo texto de entrada puede ser cualquiera siempre y cuando el tamaño de dicho archivo texto no sea inferior a 10MB (busque un texto en Internet, EN ESPAÑOL, puede ser El Quijote por ejemplo, que es de uso libre, que tengan puntos aparte y puntos seguidos, números y palabras terminadas en n, cópielo y úselo como entrada a su programa).

NOTA4: Recuerde... PERFORMANCE, performance, performance.