

Segundo Proyecto Sistemas Operativos

Compresor Distribuido
Sistemas Operativos
II Semestre 2024

1. Motivación

El propósito de este proyecto es darle familiaridad con el manejo básico de POSIX y sistemas de archivos, esto aunado al poder manejar un sistema de compresión y paralelizar sus procesos.

2. Objetivos Formativos

2.1. Objetivo(s) del curso que respalda el proyecto

1. Proporcionar una comprensión sólida de la teoría y práctica básica de los Sistemas Operativos (objetivo principal).
2. Conocer el funcionamiento de los Sistemas Operativos.
3. Resolver problemas clásicos de los Sistemas Operativos.

2.2. Contenido(s) del curso que cubre el proyecto

- Conceptos Básicos
 - Llamadas al sistema
- Administración de Procesador
 - Conceptos
 - Comunicación entre procesos
 - Threads
 - Planificación del CPU
 - Sincronización de Procesos
 - Bloqueos

- Administración de dispositivos

Hardware
Interfaz
Desempeño

- Sistemas Distribuidos

Estructura de Redes
Comunicación Distribuida
Coordinación Distribuida
Sistema de Archivos Distribuido

3. Especificación del proyecto

El proyecto consiste en realizar un compresor que aproveche varios núcleos para comprimir un archivo, utilizando árboles de Huffman.

3.1. El algoritmo de Huffman

El algoritmo consiste en la creación de un árbol binario que tiene cada uno de los símbolos por hoja, y construido de tal forma que siguiéndolo desde la raíz a cada una de sus hojas se obtiene el código Huffman asociado a él.

1. Se crean varios árboles, uno por cada uno de los símbolos del alfabeto (en nuestro caso siempre el alfabeto será un byte), consistiendo cada uno de los árboles en un nodo sin hijos, y etiquetado cada uno con su símbolo asociado y su frecuencia de aparición.
2. Se toman los dos árboles de menor frecuencia, y se unen creando un nuevo árbol. La

etiqueta de la raíz será la suma de las frecuencias de las raíces de los dos árboles que se unen, y cada uno de estos árboles será un hijo del nuevo árbol. También se etiquetan las dos ramas del nuevo árbol: con un 0 la de la izquierda, y con un 1 la de la derecha.

3. Se repite el paso 2 hasta que solo quede un único árbol. Con este árbol se puede conocer el código asociado a un símbolo, así como obtener el símbolo asociado a un determinado código.

Luego cada código consiste en la ruta (1 para izquierda y 0 para derecha, o al revés) de cada hoja desde la raíz. El que sea un árbol garantiza que se trata de un código único para cada hoja, que representa el símbolo del alfabeto correspondiente.

Luego cada símbolo es reemplazado por su secuencia única (o código de Huffman) en un archivo.

3.2. Paralelización en la compresión

Su programa debe distribuir el conteo de frecuencias, para ello, debe partir el archivo en n partes, siendo n el número de servidores que se hayan configurado, les envía por socket $\frac{t}{n}$ bytes a cada servidor (aproximadamente) siendo t el tamaño de bytes del archivo.

Cada servidor debe retornar una tabla de 256 enteros con las frecuencias de su sección de archivo, se genera una tabla consolidada y se genera el árbol (utilizando una cola de prioridad para encontrar los dos nodos menos frecuentes). Este árbol debe ser enviado de vuelta a los servidores para que cada uno de ellos reemplacen los pedazos de archivos enviados previamente con los códigos de Huffman producidos por el árbol.

Finalmente, se reciben todos los bits de las partes comprimidas, se consolidan (y se pegan) en un sólo archivo y se guarda la tabla de códigos de huffman.

3.3. Descompresión secuencial

Dado el archivo comprimido y la tabla de códigos, se debe poder descomprimir el archivo ori-

ginal.

3.4. Entrada de los programas

El compresor va a recibir por un archivo de configuración, cuyo criterio queda sujeto a cada grupo de trabajo, las IP's de los servidores que cuentan frecuencias y comprimen, el nombre y ubicación del archivo a comprimir, el nombre del archivo que va a quedar comprimido y el nombre del archivo donde se guardará la tabla de huffman.

El descompresor recibe el archivo comprimido y el archivo con la tabla de huffman y el nombre del archivo donde se guardará el archivo descomprimido.

4. Metodología

Los estudiantes se deben sentar para diseñar la solución del proyecto y luego se programa.

5. Rúbrica

Si el descompresor no deja el archivo exactamente igual que el original, se otorgará una nota de cero.

Producto esperado	Porcentaje
Partición del archivo	15 %
Distribución del archivo	15 %
Creación del árbol	15 %
Códigos de Huffman	15 %
Compresión distribuida	15 %
Compactación de la compresión	25 %
TOTAL	100 %

Para la rúbrica de la documentación, se solicita que sea en Latex, si no se cumple esta premisa, se otorgará un cero.

Producto esperado	Porcentaje
Portada	5 %
Introducción	5 %
Marco Teórico	15 %
Descripción de los resultados	40 %
Conclusiones	15 %
Descripción de la experiencia	15 %
Aprendizajes (personal)	5 %
TOTAL	100 %

6. Estimación de tiempo

- El proyecto puede ser realizado en grupos de a lo sumo 4 personas
- Fecha de entrega: 25 de octubre.
- Todos los miembros del equipo de trabajo deben ser capaces de explicar todos los módulos del proyecto.

7. Aspecto Generales

- Formato de entrega: PDF y código al correo `edramirez@itcr.ac.cr`
- Se asignarán citas de revisión, pero el programa debe correr en la computadora del profesor (linux estándar) o en las computadoras del LAIIMI.
- Formato del asunto indicado en la carta al estudiante IC-6600 SO Proyecto1