

**Evaluación, Unidad 3 (40%)**  
**(Índice Invertido)**

Fecha de Entrega	22/Junio/2025
Carrera	Ingeniería Civil Informática
Académico	Ricardo J. Barrientos Rojel
Actividad Curricular	Sistemas Distribuidos (INF-515 – S2)

Resultados de Aprendizajes Evaluados:	Implementar aplicaciones distribuidas, haciendo uso de computación de alto desempeño en problemas científicos y prácticos, fomentando el trabajo en equipo.
---------------------------------------	---

PUNTAJE MÁXIMO	6,0	PUNTAJE DE CORTE	3,6
----------------	-----	------------------	-----

INSTRUCCIONES
<ul style="list-style-type: none"><li>○ Debe desarrollar esta tarea en pareja. Debe realizar la tarea con la misma pareja que en la Evaluación de la Unidad 2 (k-means). Si cambia de pareja, avisar por email a <a href="mailto:rbarrientos@ucm.cl">rbarrientos@ucm.cl</a> con quien trabajará, a más tardar el <b>10 de Junio</b>.</li><li>○ Plazo de entrega: Por LMS a más tardar el <b>22 de Junio del 2025</b>.</li></ul> <p><b><u>Restricciones</u></b> Se aplicará artículo 67º del reglamento del estudiante, el cual indica que, en caso de sorprender copia parcial o exacta, ya sea entre compañeros o reproducidos de algún medio, lo cual implica un 1,0 para todos los involucrados.</p>

## Descripción

Este ejercicio consiste en realizar una implementación básica de un **Índice Invertido** (usando MPI en lenguaje C o C++), que es la estructura de datos utilizada como base por los motores de búsqueda Web para procesar sus consultas. La creación del índice debe ser implementada en **paralelo**, usando múltiples procesos en MPI sobre una máquina multi-núcleo. Posteriormente, para comprobar el correcto funcionamiento del índice, usted deberá procesar consultas en él.

Un índice invertido consiste en un conjunto de términos, y por cada término se tiene asociada una lista de documentos, que son los documentos donde dicho término aparece. Por ejemplo, considere que se tienen los siguientes 3 archivos HTML con los siguientes términos en ellos:

*primero.html:*  
hola mundo perro

*segundo.html:*  
hola perro gato auto

*tercero.html:*  
perro auto casa

El procesar dichos archivos debe generar un índice invertido como indica la Figura 3.7. En este ejemplo cada término tiene asociado una lista enlazada de documentos, que contiene los nombres de los archivos donde aparece dicho término. Con lo anterior, si un usuario desea realizar la consulta “auto”, su sistema debería entregar como respuesta los documentos “segundo.html” y “tercero.html”, porque en esos documentos es donde el término “auto” está presente.

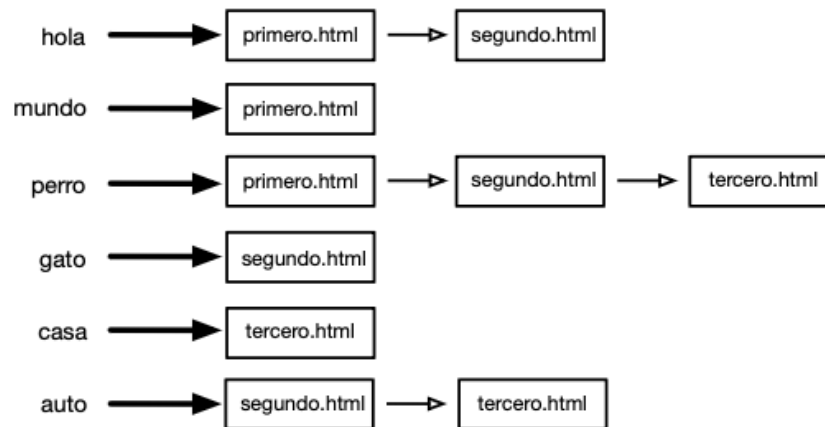


Figura 3.7: Ejemplo de Índice Invertido.

Se utilizará un archivo de entrada para ingresar los documentos, y también las consultas, redirigiendo la entrada de los datos con “<” (Ej.: `mpirun -np 8 ./a.out < input.txt`). Para el ejemplo anterior, el archivo de entrada `input.txt` tendría el siguiente formato:

```
3
primero.html
3
hola mundo perro
segundo.html
4
hola perro gato auto
tercero.html
3
perro auto casa
5
gato perro mundo edificio hotel
```

En donde la primera línea indica el número de documentos, la segunda línea el título del primer documento, la tercera línea el número de términos del primer documento, la cuarta línea indica los términos del primer documento, y se repite lo mismo para los siguientes 2 archivos. Las últimas 2 líneas indican el número de consultas y las consultas mismas, que están compuestas por sólo 1 término cada una. En este ejemplo, las consultas son 5: “gato”, “perro”, “mundo”, “edificio” y “hotel”. Fíjese que es posible que una consulta no tenga archivo alguno como resultado.

Debe definir el número de procesos a utilizar como una constante dentro de su programa, de tal manera que se pueda modificar de ser necesario. Al momento de resolver consultas, su programa debe imprimir por pantalla, de manera muy clara, cuales son los nombres de archivo que representan el resultado de cada consulta. No importa el orden en que aparecen los resultados de las consultas, solo importa que el resultado sea el correcto. En caso que no haya ningún resultado para una consulta, debe indicarlo por pantalla también. Por ejemplo, de la siguiente manera:

```
Resultados para "gato": segundo.html
Resultados para "perro": primero.html segundo.html tercero.html
Resultados para "mundo": primero.html
Resultados para "edificio": No hay resultados.
Resultados para "hotel": No hay resultados.
```

**Todos** los procesos deben involucrarse en la creación del índice invertido. Usted decida cómo distribuye el trabajo entre los procesos, pero **todos** ellos deben colaborar. Por ejemplo, podría decidir que cada lista de documentos sea creada por un proceso distinto, como muestra la Figura 3.8 utilizando 3 procesos.



Figura 3.8: Ejemplo de distribución posible para un Índice Invertido.

Una vez que el índice invertido está creado, debe implementar el resolver consultas. Por ejemplo, para procesar una consulta usando el proceso de creación del índice invertido del ejemplo anterior, el proceso 0 puede leer todas las consultas y enviarlas al proceso que tiene el término de la consulta, para que dicho proceso responda con la respuesta correcta.

Para calificar su tarea, se tomará en cuenta los siguientes ítems:

- 1) Correcta lectura de los datos. (0,5 pts.)
- 2) Correcta estrategia de distribución del cómputo (3,0 pts.)
- 3) Correcto uso de MPI y sus funciones (1,5 pts.)
- 4) Impresión correcta de resultados. (1,0 pts.)

**Nota:**

- Esta tarea será revisada con conjunto con los estudiantes de cada grupo. El grupo deberá responder a las dudas planteadas en la revisión, de no hacerlo tendrá -3,0 unidades de nota.