

Informe Caso 2 InfraComp

- **Estructuras de datos usadas**

Para la simulación del comportamiento del sistema de paginación, recurrimos al uso de dos estructuras de datos; una para los marcos de página y otra para la tabla de páginas. Para los marcos de página se usó un contenedor simple del tamaño del número de marcos. Por otro lado, para la tabla de páginas se usó una matriz de tamaño $[n][3]$, n siendo el número de páginas y las 3 columnas para el número de página, su bit R y su bit M. La tabla de páginas actualiza sus bits R y M en los siguientes casos: con las referencias generadas por el proceso y cuando ocurre una interrupción de reloj. En cuanto a los marcos de página, se actualizan cuando una referencia de un proceso ocasiona un remplazo de página, en este caso se actualiza uno de los marcos de acuerdo con el algoritmo de LRU.

- **Esquema de sincronización**

Para la sincronización, decidimos usar sleep y tener métodos específicos en synchronized. Estos métodos son aquellos que acceden a los bits de R o M de la tabla de páginas, ya sea para escritura o lectura. Nos aseguramos de que los sleep no estén dentro de métodos synchronized porque así evitamos dormir el proceso completo. Por esto, los sleep están desincronizados.

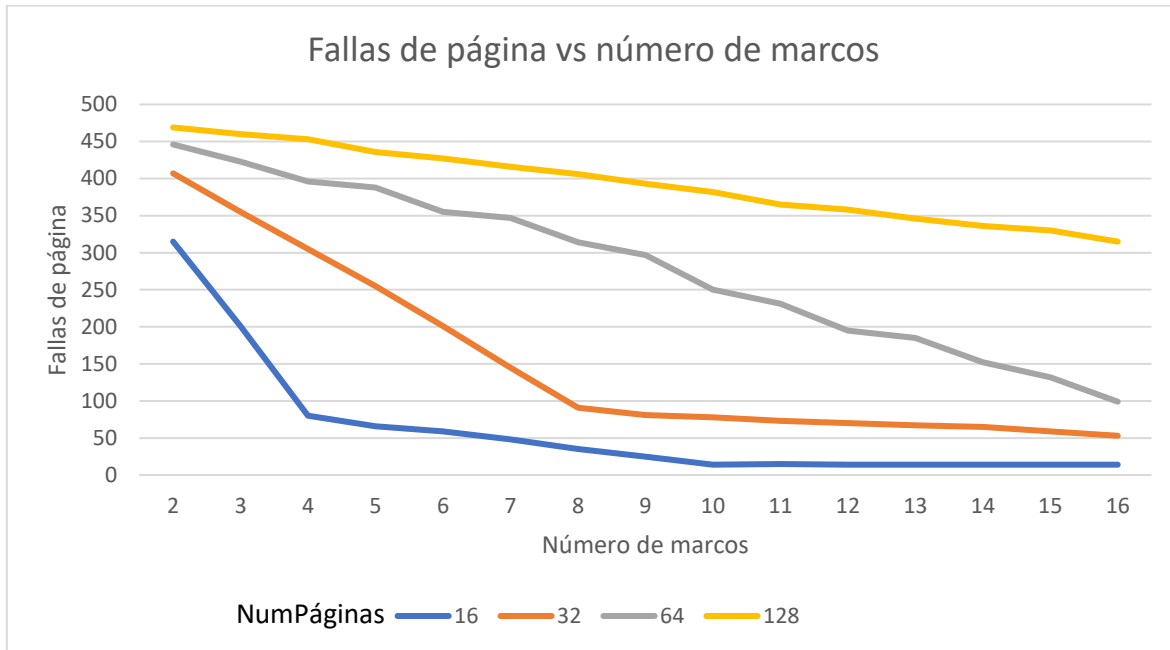
- **Tabla de datos**

	Fallas de página		
NúmPáginas	Intento 1	Intento 2	Intento 3
16	34	36	35
32	87	86	91
64	313	315	314
128	406	406	406

- **Varianza de fallas**

El número de fallas puede variar debido a que el thread cíclico que reinicia el valor de la R puede activarse antes de una referencia en la ejecución de un proceso, pero al volverse a ejecutar, puede activarse después. Esto puede afectar el número de fallas ya que la página a remplazar puede variar.

- **Gráfica**



Los resultados de la gráfica tienen sentido con el sistema de paginación simulado, ya que se evidencia una relación inversamente proporcional entre el número de marcos y las fallas de página. Es decir, mientras más marcos, menos fallas de página, lo que tiene sentido ya que, al darle más espacio en memoria al programa, este requiere hacer menos remplazos de página.

Además, se evidencia que mientras más grande el número de páginas, mayores las fallas de página, lo cual tiene sentido, ya que, para un mismo espacio en memoria, si se tiene un tamaño de programa mayor, es esperado que se tengan más fallas.

Por último, se puede evidenciar que cuando el número de marcos es mayor o igual a la cantidad de páginas del proceso, la cantidad de fallos será menor o igual al número de páginas del proceso (dependiendo de las que se referencien).