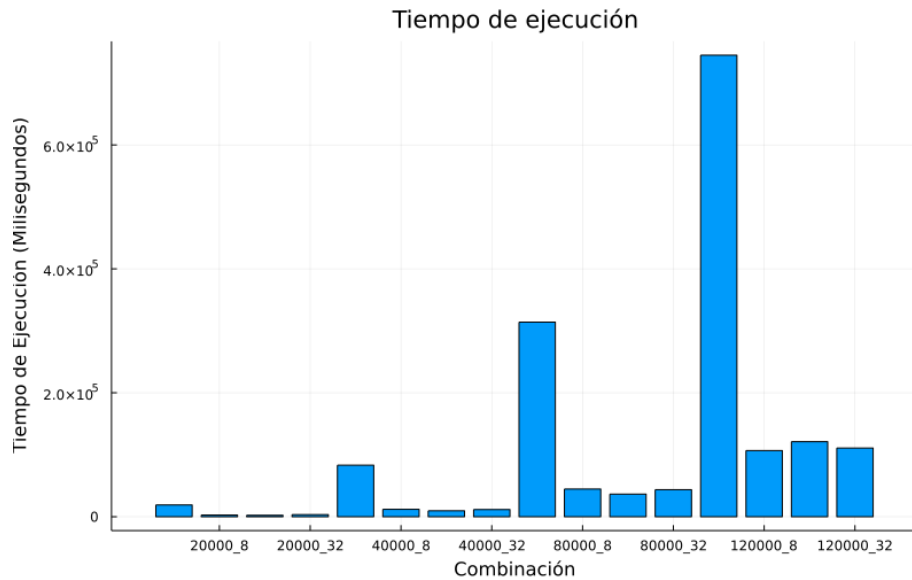


# Reporte experimental - DBSCAN y Opem MP

Héctor G. T. Torres - 191589

## Resultados

A continuación, se muestran las gráficas con los tiempos de ejecución de cada permutación de número de hilos y cantidad de puntos. Cada gráfica se representa en formato de **número de puntos\_número de hilos**. Por ejemplo, la barra que corresponde a la permutación de 20000 puntos y 8 hilos es la 20000\_8.



Como se puede ver en las gráficas, la mejora que representa la paralelización presenta rendimientos decrecientes: la mejora en rendimiento obtenida de pasar de 1 a 8 hilos es menor en porcentaje que la obtenida al pasar de 8 a 16 hilos. Esto se vuelve aparente al ver la diferencia entre usar 1 y 8 hilos. En esos casos, la paralelización hizo que el algoritmo se ejecutara entre 6.5 y 7.0 veces más rápido.

Aunado a esto, se puede observar que no hay mejora significativa al pasar de 16 (el máximo de hilos) a 32 hilos (el doble del máximo), de hecho hasta empeora un poco. Probablemente esto se debe a que la librería `omp.h` toma como valor predeterminado el número de hilos máximo si se ingresa un número mayor a

este.

En cuanto al número de puntos, se puede observar que se mantuvo una proporción de mejora similar entre los números de puntos en cada conjunto de datos. Es decir, sin importar el número de puntos en el *dataset*, la implementación del algoritmo mejoraba en proporciones similares conforme aumentaba el número de hilos.