

# Reflexión actividad integral 3.3

A00838495 Rodrigo Martínez Escalante

En la actividad que realizamos hay una razón principal por la cual usamos binary heap en lugar de una búsqueda binaria es porque no solo estamos buscando valores, si no que nos importan los valores más altos. En este caso, es mucho más sencillo solo hacer un top() 10 veces y obtener la lista con los valores más altos, mientras que en la búsqueda binaria de árbol, si no está balanceado, tendríamos que hacer muchas más iteraciones.

En general, heap es muy eficiente cuando los valores más relevantes son los más altos, por la misma naturaleza del algoritmo, al siempre saber donde está el valor más alto, y su capacidad para acomodarse tras cada iteración.

Uno de los casos más comunes para usar heap es en la distribución de recursos de una computadora para ejecutar operaciones. La idea general es que se priorizan las operaciones más críticas, a pesar de que se usen estrategias para no saturar los recursos con tareas críticas y muy largas, la base de la asignación de recursos sí se define por hacer lo más crítico primero, y heap es muy eficiente para esos casos.