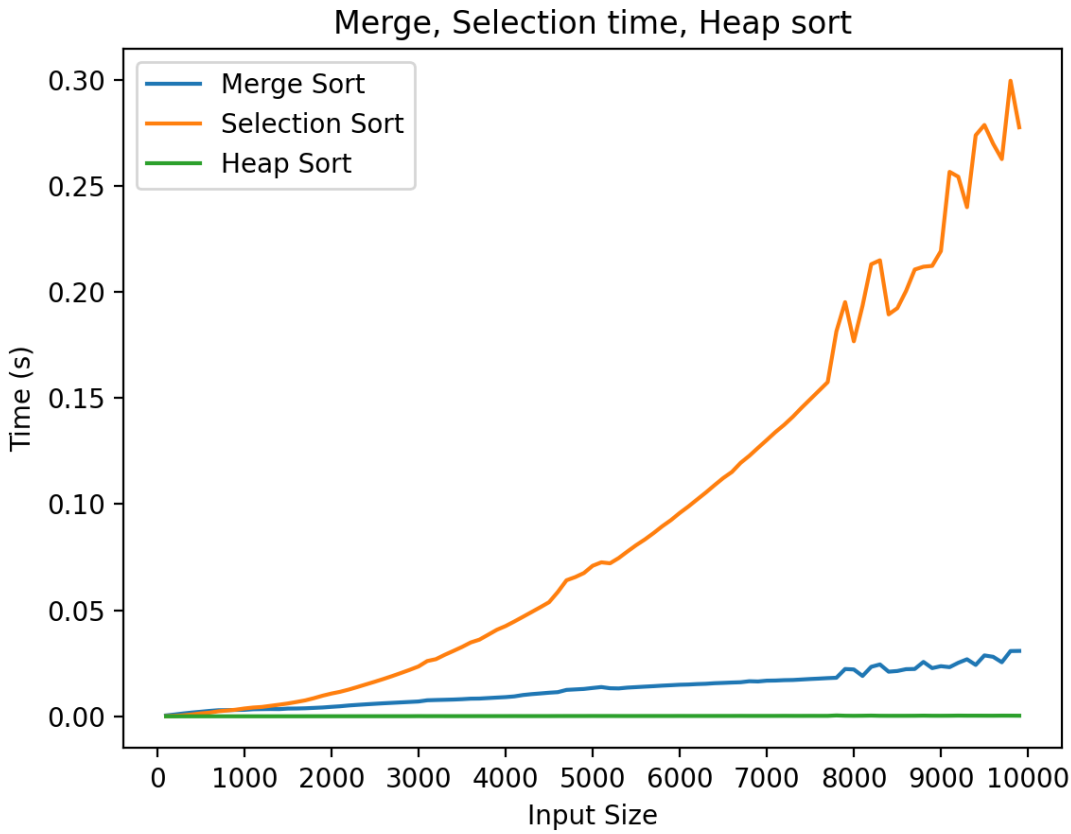
CS271 Project 3

Wilson Le, Tung Luu

October 2021



- Asymptotic time complexity of the heap sort algorithm on an array that is already sorted:
  - $\Theta(n \log n)$
- Asymptotic time complexity on an array that is in reverse order:
  - $\Theta(n \log n)$
- Best case asymptotic time complexity of heap sort
  - $\Theta(n)$
- What kind of input does the best case asymptotic time complexity occur
  - The best case asymptotic time complexity of heap sort will be $\Theta(1)$ in the case that all elements in the input array are the same. First, the buildHeap method will take $\Theta(n)$ when we initialize the heap. When we call heapSort method, since all elements are the same, the heapify method will take $\Theta(1)$ in each iteration of n iteration, since heapify doesn't need to move any element down the heap. Therefore the asymptotic time complexity of heap sort in this case is $\Theta(1)$.