Example PDF

Merge Sort

- prerequisites : recursion + divide and conquer
- stable
- not in place
- auxilary space requirements : O(n) because of auxilary left and right subarrays

...

Time Complexity Analysis

:

$$T(n) = 2T(n/2) + \Theta(n)$$

:

$$\Theta(n \log n)$$

Code in Python

```
MergeSort(arr):
  if len(arr) > 1:
    mid = len(arr)//2     # create left and
    left = arr[:mid]        right subarrays
    right

    ...
```