SUMMARY OF SORTING ALGORITHMS

Name	Category	Method/ Working	Stable ?	In place?	Running Time	Applicability/Suitability	Known Applications	Remarks
Selection Sort	Internal Comparison Based	Selection	N	Y	O(n^2)	For small input, Generally, not used in practical applications due to quadratic time		-
Bubble sort	Internal Comparison Based	Exchange	Y	Y	Best - O(n) (For optimized algo.) Average - O(n^2) Worst- O(n^2)	For small input, If the input is already sorted, then optimized bubble sort works well. Generally, not used in practical applications due to quadratic time		-
Insertion sort	Internal Comparison Based	Insertion	Y	Y	Best - O(n) Average - O(n^2) Worst- O(n^2)	Algorithm works well if input is almost sorted. This requires lesser number of shift operations	1. Timsort (used in python library funtions) is a hybrid of mergesort and insertion sort 2. Introsort (used in C++, Java library functions) is a hybrid of quick sort, insertion sort and Heap Sort	-
Shell Sort	Internal Comparison Based	Insertion	N	Y	Best - O(n log n) Average - O($n^{3/2}$) Worst- O(n^2) (for original gap sequence of N/2^K)	Performs more number of swap operations than insertion sort, hence its more costly	Some implementations of the qsort function in the C standard library targeted at embedded systems use it instead of quicksort	-Improvement of insertion sort -Various Gap sequences were proposed- https://en.wikipedia.org/wiki/Shellsort#Gap_sequences This lets an element take "bigger steps" toward its expected position. Multiple passes over the data are taken with smaller and smaller gap sizes. The last step of Shell sort is a plain

								insertion sort, but by then, the array of data is guaranteed to be almost sorted.
Merge Sort	Internal Comparison Based (Has a external sort variant)	Merging	Y	N (Requires auxiliary array of O(n) size for merging)	O (n log$_2$n)	Can be used when worst case performance is important And memory is not a concern	Timsort is a hybrid of mergesort and insertion sort	Linked list-based implementation is in place.
Quick Sort	Internal Comparison Based	Partitioning	N	Y (O(log n)- asymptotically considered as in-place)	Best -O(n log$_2$n) Average -O(n log$_2$n) Worst- O(n^2)	Can be used when worst case performance is not much important but require a descent average case performance. In general, Quicksort is much faster than Mergesort except for worst case that can be worked around by using a good pivot.	Introsort is a hybrid of quick sort, insertion sort and Heap Sort Used in Qsort() function in C library with aome other alternative also	**Variants** 1. last element as pivot 2. Random element as pivot 3. Middle element as pivot 4. Median as a pivot 5. Element >= (n/4) elements and <= (n/4) elements
Heap Sort	Internal Comparison Based	Selection	N	Y	O(n log$_2$n)	When worst case performance is important and memory is also limited. However, it is slower than quicksort and Mergesort due to a lot of swap operations. So, it is not used alone, in practical applications.	Introsort is a hybrid of quick sort, insertion sort and Heap Sort	
Counting Sort	Internal Non-comparison Based	--	Y	N (n+r, where r is the number range)		When the range of keys (numbers) in a list is small.	-	
DO IT YOURSELF								
Radix Sort								

TimSort							Python sort(), sorted()	
In Place MergeSort								
Introsort							Used in C++ STL, GNU C++ library, Java library sort functions.	https://en.wikipedia.org/wiki/Introsort

NOTE :

1. Radix Sort, TimSort, Introsort, in-place mergesort are not in syllabus.

2. You may add additional info. like year of invention and inventor name.