|  |  |
| --- | --- |
| Bubble Sort |  |
|  |  |
| Quicksort algorithm |  |
|  | **public** **class** QuickSort {    **public** **static** **void** quickSort(**int** arr[],**int** start, **int** end) {  **if**(start<end) {    **int** partIndex = *partition*(arr,start,end);  *quickSort*(arr, start,partIndex-1);  *quickSort*(arr, partIndex+1,end);  }      }  **private** **static** **int** partition(**int**[] arr, **int** start, **int** end) {    **int** pivotElement = arr[end];  **int** partIndex = start;    **for** (**int** i = start; i < end; i++) {  **if**(arr[i]<=pivotElement) {  *swap*(arr, i,partIndex);  partIndex ++;  }  }  *swap*(arr, end,partIndex);  **return** partIndex;  }  **private** **static** **void** swap(**int**[] arr, **int** i, **int** partIndex) {  **int** temp = arr[partIndex];  arr[partIndex] = arr[i];  arr[i] = temp;  }  } |
| Bucket Sort |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |