Quick Sort

Sreehari P Sreedhar, CB.SC.I5DAS20032

```
In [ ]:   # function to get at least 10 values in an array

          def getVals(arr):

              print('Current length of array:', len(arr))
              print('Current array:', arr)

              arr += (input("Enter numbers separated by spaces: ").split())
              arr = [int(i) for i in arr if i.isdigit()]

              print('Current length of array:', len(arr))
              print('Current array:', arr)

              if len(arr) < 10:
                  arr = getVals(arr)

              return arr
```

```
In [ ]:   def split(arr, l, r):
              i = l
              pivot = arr[l]

              while l < r:
                  while l < len(arr) and arr[l] <= pivot: # find first element greater than 
                      l += 1

                  while arr[r] > pivot: # find first element less than pivot
                      r -= 1

                  if l < r:
                      arr[l], arr[r] = arr[r], arr[l] # swap elements

              arr[r], arr[i] = arr[i], arr[r] # swap pivot with first element greater than p

              return r
```

```
In [ ]:   # function to perform quick sort on array
          def quickSort(arr, l, r):
              if l < r:
                  splitPoint = split(arr, l, r) # split the array and return the index of the

                  quickSort(arr, l, splitPoint - 1) # sort the left half of the array
                  quickSort(arr, splitPoint + 1, r) # sort the right half of the array

              return arr
```

```
In [ ]:   arr = getVals([])

          quickSort(arr, 0, len(arr) - 1)
```

```
Current length of array: 0
Current array: []
Current length of array: 12
Current array: [3, 43, 54, 234, 236, 56, 22, 43, 65, 98, 87, 876]
```
Out[ ]:
```
[3, 22, 43, 43, 54, 56, 65, 87, 98, 234, 236, 876]
```

In [ ]: