Ch 7
Quicksort

Workshop12

CSC-340
April 4, 2022

Name:____________________________________

In CoCalc Workshop `7.2-Quicksort_Performance`, write a program `quicksort.cpp` OR `quicksort.py` that ultimately implements and demonstrates the QUICKSORT algorithm to sort a one-based array.

Listing 1: print a one based array

```
/*
* print_one_based(v)
*  takes integer vector v as a const reference parameter
*  Prints the contents of vector v AFTER the initial unused position
*  v is not modified
*/
```

Listing 2: partition

```
/*
* partition(A, p, r)
*    takes integer vector A as a reference parameter
*    subarray A[p..r] is the vector to partition into 3 (possibly empty) regions:
*  A[p..q-1] elements <= A[q]
*  A[q] = pivot
*  A[q+1..r] elements > A[q]
*    return q, the index of the pivot
*    A IS modified
*/
```

Listing 3: quicksort

```
/*
* quicksort(A,p,r)
*  takes integer vector A as a reference parameter
*   subarray A[p..r] is the vector to sort
*   A IS modified
*/
```

Listing 4: quicksort example run

```
One-based array A = <13,19,9,5,12,8,7,4,21,2,6,11>

After quicksort array A = <2,4,5,6,7,8,9,11,12,13,19,21>
```