**4Learning by doing-1**

Give traces by showing how the keys E A S Y Q U E S T I O N are sorted with quick sort.

* Answer the above problem in a file name problem-1.txt and add it to the folder ADS-1-practice/m11
* Push the respository to GitHub when your answer is done
* Enter the git commit ID in the blank below

Do not use eval to submit this activity

partition

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| LOW | HI | PI | I | J | E | A | S | Y | Q | U | E | S | T | I | O | N |
| 0 | 11 | N | -1 | 0 | E | A | S | Y | Q | U | E | S | T | I | O | N |
|  |  |  | 2 | 6 | E | A | E | Y | Q | U | S | S | T | I | O | N |
|  |  |  | 3 | 9 | E | A | E | I | Q | U | S | S | T | Y | O | N |
| Sort | | | | | | | | | | | | | | | | |
|  |  | Pi=4 | I | J | E | A | E | I | Q | U | S | S | T | Y | O | N |
|  |  | I | -1 | 3 | E | A | E | I | Q | U | S | S | T | Y | O | N |
| Sort | | | | | | | | | | | | | | | | |
|  |  | PI=2 | I | J | E | A | E | I | Q | U | S | S | T | Y | O | N |
|  |  | A | -1 | 0 | A | E | E | I | Q | U | S | S | T | Y | O | N |
|  |  | N | -1 |  | A | E | E | I | N | U | S | S | T | Y | O | Q |
|  |  | Q | -1 | 0 | A | E | E | I | N | O | S | S | T | Y | U | Q |
|  |  | Q | -1 | 0 | A | E | E | I | N | O | Q | S | T | Y | U | S |
|  |  | S | -1 | 0 | A | E | E | I | N | O | Q | S | S | Y | U | T |
|  |  | T |  |  | A | E | E | I | N | O | Q | S | S | T | U | Y |

**Activity6**

**write the missing blanks**

**private static int partition(Comparable[ ] a, int lo, int hi) {  
       int i = lo;  
       int j = hi + 1;  
       Comparable v = a[lo];  
       while (true) {   
           while (less(a[---------], v)) { //Blank 1=a[++j]  
               if (i == --------) break; //Blank 2 = hi  
           }**

**while (less(------------, a[--j])) { // Blank 3 = V  
               if (j == lo) --------------;     // Blank 4 = break;**

**}  
           if (i >= ----------) break; //Blank 5 = j  
  
           exch(a, i, j);  
       }  
  
       exch(a,-------------, j); //Blank 6 = lo  
  
       return j;  
   }**

**Activity7**

**private static int partition(Comparable[] a, int lo, int hi) {  
       int i = lo;  
       int j = hi + 1;  
       Comparable v = a[lo];  
       while (true) {   
           while (less(a[++i], v)) {  
               if (i == hi) break;  
           }  
           while (less(v, a[--j])) {  
               if (j == lo) break;    
           }  
 if (i >= j) break;  
  
            exch(a, i, j);  
       }  
       exch(a, lo, j);  
       return j;  
   }**

**Note: output should be in the following format**

**[1, 2, 0, 3, 5, 8, 7]**

**What will be the output when6, we call partition with parameters({1, 2, 8, -1, 0, -3, 4}, 0, 6 )**

**--------------------------**

[-1,-3,8,1,0,2,4]

**What will be the output when we call partition with parameters({5, 8, 9, 10, 21, 25, 11, 13, 15, 12}, 4, 9 )**

**--------------------------[5,8,9,10,25,21,11,13,15,12]**

**What will be the output when we call partition with parameters({0, 1, 2, 3, 4}, 0, 4 )**

**--------------------------**

**[0,3,4,2,1]**

**What will be the output when i call partition with parameters({14, 6, 7, 10, 9, 8}, 0, 1 )**

**--------------------------**

**[7,6,14,10,9,8]**