**BigOAnalysis**

Pawan Kumar Bagri

Q1. O(1) + O(n) = O(n).

Q2. Complexity of recursive function O(log(end))

O(lon(n)).

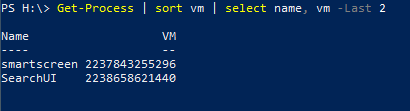
Q3. Option 4 (both 1 and 2)

Q4. [Performance Testing](https://qatestlab.com/services/test-automation/performance-testing/) is mostly about checking the system’s responsiveness under the given conditions. Its main task is to define the bottlenecks of the tested application.

Performance Engineering goes through the whole process of the Software Development Life Cycle. Its main objective is to find, design and implement the solution for the identified problem.

Q5. Reading purpose only.

Q6.



Q7.

Code 1

public class linearCode {

static void processAndPrintInfo(int[] array, int size){

System.out.printf("First element of array = %d\n", array[0]);

for(int i = 0; i < size/2; i++){

System.out.println("%d\n", array[i]);

}

for(int i=0; i<100; i++){

System.out.println("Hi");

}

}

public static void main(String[] args){

System.out.println("Hello World!");

int[] array = new int[]{1, 2, 3, 4, 5, 6};

processAndPrintInfo(array, array.length);

}

}

Code 2:

public class binarySearch {

boolean recursiveFunction(int[] arr, int x, int start, int end){

if(start > end){

return false;

}

int mid = (start + end) / 2;

if(arr[mid] == x){

return true;

}

if(arr[mid] > x){

return recursiveFunction(arr, x, start, mid-1);

else{

return recursiveFunction(arr, x, mid+1, end);

}

}

void searchValue(int[] arr, int x){

if(recursiveFunction(arr, x, 0, arr.length-1)){

System.out.println("Element found!");

}

else{

System.out.println("Element not found!");

}

}

public static void main(String args[]){

binarySearch bs = new binarySearch();

int[] arr = new int[]{2, 1, 4, 3, 6, 5};

bs.searchValue(arr, 4);

}

}