Free Response Question

Pause and write BFS based pseudocode for printing all components of an unconnected undirected graph. It will have the following two functions:

Void **BFS** (List<Node> adjList)

Void **ExploreBFS** (Node start, seen, component)

ANSWER:

There should be some **Pseudocode** below. If you don't see it, then please refresh.

Please note, that even though it says below that the language is Java, it's actually **Pseudocode**. Java is selected only to provide better syntax highlighting.

Consider this BFS pseudocode structure to print all components.

```
JAVA
Node {List<Node> neighbors; int label}

Void BFS (List<Node> adjList)
    seen: set of integers, empty initially
    for (cur in adjList)
        If (cur.label not in seen)
                component: list of integers, empty
                ExploreBFS (cur, seen, component)
                print(component)
```

```
JAVA
// print the shortest distance between start and target, in terms of number of edges
void ExploreBFS (Node start, seen, component)
    bfs_q: fifo  queue of node labels, empty initially
    bfs_q.add(start.label)
    seen.add (start.label)
    component.add(start.label)
```

```
    while (bfs_q not empty)
        cur = bfs_q.remove
        for (next: cur.neighbors)
                if (next is not seen)
                        bfs_q.add(next.label)
                        seen.add(next.label)
                        component.add(next.label)

    Print component and return
```