

DISJOINT-SET

I. Java Implementation

goodValues

```
boolean goodValues() {
    boolean hasRoot = false;
    for (int i = 0; i < s.length; i++) {
        if (s[i] == -1) {
            hasRoot = true;
        } else if (!(s[i] >= 0 && s[i] < s.length)) {
            return false;
        }
    }
    return hasRoot;
}
```

acyclic

```
boolean acyclic() {
    boolean[] visited = new boolean[s.length];
    for (int i = 0; i < visited.length; i++) {
        visited[i] = false;
    }
    int[] parents = new int[s.length];
    for (int i = 0; i < s.length; i++) {
        if (!visited[i]) {
            int numParents = 0;
            int currentIndex = i;
            while (s[currentIndex] != -1) {
                for (int j = 0; j < numParents - 1; j++) {
                    if (parents[j] == s[currentIndex]) {
                        return false;
                    }
                }
                parents[numParents] = s[currentIndex];
                numParents++;
                visited[currentIndex] = true;
                currentIndex = s[currentIndex];
            }
        }
    }
    return true;
}
```

}