

Code - class Solution {

public:

```
#define NO_CAMERA      0
#define HAS_CAMERA     2
#define NOT_NEEDED     1
int ans = 0;
int dfs(TreeNode *root) {
    if (!root) return NOT_NEEDED;
    int l = dfs(root->left);
    int r = dfs(root->right);
    if (l == NO_CAMERA || r == NO_CAMERA) {
        ans++;
        return HAS_CAMERA;
    } else if (l == HAS_CAMERA || r == HAS_CAMERA) {
        return NOT_NEEDED;
    } else {
        return NO_CAMERA;
    }
}
int minCameraCover(TreeNode* root) {
    if (dfs(root) == NO_CAMERA) ans++;
    return ans;
}
};
```

Time complexity – $O(N)$

Space complexity – $O(N)$

