Java

```java
class Solution {
    public String findSmallestRegion(List<List<String>> regions, String region1, String region2) {

    }
}
```

JavaScript

```javascript
/**
 * @param {string[][]} regions
 * @param {string} region1
 * @param {string} region2
 * @return {string}
 */
var findSmallestRegion = function(regions, region1, region2) {

};
```

TypeScript

```typescript
function findSmallestRegion(regions: string[][], region1: string, region2: string): string {

};
```

C++

```
class Solution {
public:
    string findSmallestRegion(vector<vector<string>>& regions, string region1, string region2) {

    }
};
```

C#

```
public class Solution {
    public string FindSmallestRegion(IList<IList<string>> regions, string region1, string region2) {

    }
}
```

Kotlin

```
class Solution {
    fun findSmallestRegion(regions: List<List<String>>, region1: String, region2: String): String {

    }
}
```

Go

```
func findSmallestRegion(regions [][]string, region1 string, region2 string) string {

}
```
