

Java

```
class Solution {  
    public List<List<Integer>> getAncestors(int n, int[][] edges) {  
  
    }  
}
```

JavaScript

```
/**  
 * @param {number} n  
 * @param {number[][]} edges  
 * @return {number[][]}  
 */  
var getAncestors = function(n, edges) {  
  
};
```

TypeScript

```
function getAncestors(n: number, edges: number[][]): number[][] {  
  
};
```

C++

```
class Solution {
```

```
public:
    vector<vector<int>> getAncestors(int n, vector<vector<int>>& edges) {

    }
};
```

C#

```
public class Solution {
    public IList<IList<int>> GetAncestors(int n, int[][] edges) {

    }
}
```

Kotlin

```
class Solution {
    fun getAncestors(n: Int, edges: Array<IntArray>): List<List<Int>> {

    }
}
```

Go

```
func getAncestors(n int, edges [][]int) [][]int {

}
```
