

Java

```
class Solution {  
    public int[] minimumTime(int n, int[][] edges, int[] disappear) {  
  
    }  
}
```

JavaScript

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

TypeScript

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

C++

```
class Solution {
public:
    vector<int> minimumTime(int n, vector<vector<int>>& edges, vector<int>& disappear) {

    }
};
```

C#

```
public class Solution {
    public int[] MinimumTime(int n, int[][] edges, int[] disappear) {

    }
}
```

Kotlin

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

    }
}
```
