

Java

```
class Solution {  
    public int minCost(int n, int[][] edges, int k) {  
  
    }  
}
```

JavaScript

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

TypeScript

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

C++

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

    }
};
```

C#

```
public class Solution {
    public int MinCost(int n, int[][] edges, int k) {

    }
}
```

Kotlin

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

    }
}
```

Go

```
func minCost(n int, edges [][]int, k int) int {

}
```
