Java

```java
class Solution {
    public int shortestPathWithHops(int n, int[][] edges, int s, int d, int k) {
        
    }
}
```

JavaScript

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

TypeScript

```typescript
function shortestPathWithHops(n: number, edges: number[][], s: number, d: number, k: number): number {
    
};
```

C++

```
class Solution {
public:
    int shortestPathWithHops(int n, vector<vector<int>>& edges, int s, int d, int k) {
        
    }
};
```

C#

```
public class Solution {
    public int ShortestPathWithHops(int n, int[][] edges, int s, int d, int k) {
        
    }
}
```

Kotlin

```
class Solution {
    fun shortestPathWithHops(n: Int, edges: Array<IntArray>, s: Int, d: Int, k: Int): Int {
        
    }
}
```

Go

```
func shortestPathWithHops(n int, edges [][]int, s int, d int, k int) int {
```

```
}
----------------------------------------------------------------
```