**Java**

```java
class Solution {
    public boolean findSafeWalk(List<List<Integer>> grid, int health) {

    }
}
```

---

**JavaScript**

```javascript
/**
 * @param {number[][]} grid
 * @param {number} health
 * @return {boolean}
 */
var findSafeWalk = function(grid, health) {

};
```

---

**TypeScript**

```typescript
function findSafeWalk(grid: number[][], health: number): boolean {

};
```

---

**C++**

```cpp
class Solution {
```

```
public:
    bool findSafeWalk(vector<vector<int>>& grid, int health) {

    }
};
```

---

**C#**

```
public class Solution {
    public bool FindSafeWalk(IList<IList<int>> grid, int health) {

    }
}
```

---

**Kotlin**

```
class Solution {
    fun findSafeWalk(grid: List<List<Int>>, health: Int): Boolean {

    }
}
```

---

**Go**

```
func findSafeWalk(grid [][]int, health int) bool {

}
```

---