Java

```java
class Robot {

    public Robot(int width, int height) {

    }

    public void step(int num) {

    }

    public int[] getPos() {

    }

    public String getDir() {

    }
}

/**
 * Your Robot object will be instantiated and called as such:
 * Robot obj = new Robot(width, height);
 * obj.step(num);
 * int[] param_2 = obj.getPos();
 * String param_3 = obj.getDir();
 */
```

JavaScript

```javascript
/**
 * @param {number} width
 * @param {number} height
 */
var Robot = function(width, height) {

};

/**
 * @param {number} num
 * @return {void}
 */
Robot.prototype.step = function(num) {

};

/**
 * @return {number[]}
 */
Robot.prototype.getPos = function() {

};

/**
 * @return {string}
 */
Robot.prototype.getDir = function() {

};

/**
 * Your Robot object will be instantiated and called as such:
```

```
 * var obj = new Robot(width, height)
 * obj.step(num)
 * var param_2 = obj.getPos()
 * var param_3 = obj.getDir()
 */
```

TypeScript

```
class Robot {
    constructor(width: number, height: number) {

    }

    step(num: number): void {

    }

    getPos(): number[] {

    }

    getDir(): string {

    }
}

/**
 * Your Robot object will be instantiated and called as such:
 * var obj = new Robot(width, height)
 * obj.step(num)
 * var param_2 = obj.getPos()
```

```
 * var param_3 = obj.getDir()
 */
```

C++

```
class Robot {
public:
    Robot(int width, int height) {

    }

    void step(int num) {

    }

    vector<int> getPos() {

    }

    string getDir() {

    }
};

/**
 * Your Robot object will be instantiated and called as such:
 * Robot* obj = new Robot(width, height);
 * obj->step(num);
 * vector<int> param_2 = obj->getPos();
 * string param_3 = obj->getDir();
 */
```

C#

```
public class Robot {

    public Robot(int width, int height) {

    }

    public void Step(int num) {

    }

    public int[] GetPos() {

    }

    public string GetDir() {

    }
}

/**
 * Your Robot object will be instantiated and called as such:
 * Robot obj = new Robot(width, height);
 * obj.Step(num);
 * int[] param_2 = obj.GetPos();
 * string param_3 = obj.GetDir();
 */
```

Kotlin

```kotlin
class Robot(width: Int, height: Int) {

    fun step(num: Int) {

    }

    fun getPos(): IntArray {

    }

    fun getDir(): String {

    }

}

/**
 * Your Robot object will be instantiated and called as such:
 * var obj = Robot(width, height)
 * obj.step(num)
 * var param_2 = obj.getPos()
 * var param_3 = obj.getDir()
 */
```

Go

```go
type Robot struct {

}
```

```go
func Constructor(width int, height int) Robot {

}


func (this *Robot) Step(num int)  {

}


func (this *Robot) GetPos() []int {

}


func (this *Robot) GetDir() string {

}


/**
 * Your Robot object will be instantiated and called as such:
 * obj := Constructor(width, height);
 * obj.Step(num);
 * param_2 := obj.GetPos();
 * param_3 := obj.GetDir();
 */
```
