Country Road

Source code

```
class ParkingLot extends Level
{
	initialize()
	{
		//Set level size
		this.setSize(16);

		//Add borders
		this.add(new WoodenFence(0, 0));
		this.add(new WoodenFence(1, 0));
		this.add(new WoodenFence(2, 0));
		this.add(new WoodenFence(3, 0));
		this.add(new WoodenFence(4, 0));
		this.add(new WoodenFence(5, 0));
		this.add(new WoodenFence(6, 0));
		this.add(new WoodenFence(7, 0));
		this.add(new WoodenFence(8, 0));
		this.add(new WoodenFence(9, 0));
		this.add(new WoodenFence(10, 0));
		this.add(new WoodenFence(11, 0));
		this.add(new WoodenFence(12, 0));
		this.add(new WoodenFence(13, 0));
		this.add(new WoodenFence(14, 0));
		this.add(new WoodenFence(15, 0));
		this.add(new WoodenFence(15, 1));
		this.add(new WoodenFence(15, 2));
		this.add(new WoodenFence(15, 3));
		this.add(new WoodenFence(15, 4));
		this.add(new WoodenFence(15, 5));
		this.add(new WoodenFence(15, 6));
		this.add(new woodenFence(15, 7));
		this.add(new woodenFence(15, 8));
		this.add(new WoodenFence(15, 9));
		this.add(new WoodenFence(15, 10));
		this.add(new WoodenFence(15, 11));
		this.add(new WoodenFence(15, 12));
		this.add(new WoodenFence(15, 13));
		this.add(new WoodenFence(15, 14));
		this.add(new WoodenFence(15, 15));
		this.add(new WoodenFence(14, 15));
		this.add(new WoodenFence(13, 15));
		this.add(new WoodenFence(12, 15));
		this.add(new WoodenFence(11, 15));
		this.add(new WoodenFence(10, 15));
		this.add(new WoodenFence(9, 15));
		this.add(new WoodenFence(8, 15));
		this.add(new WoodenFence(7, 15));
		this.add(new WoodenFence(6, 15));
		this.add(new WoodenFence(5, 15));
```

```
            this.add(new WoodenFence(4, 15));
            this.add(new WoodenFence(3, 15));
            this.add(new WoodenFence(2, 15));
            this.add(new WoodenFence(1, 15));
            this.add(new WoodenFence(0, 15));
            this.add(new WoodenFence(0, 14));
            this.add(new WoodenFence(0, 13));
            this.add(new WoodenFence(0, 12));
            this.add(new WoodenFence(0, 11));
            this.add(new WoodenFence(0, 10));
            this.add(new WoodenFence(0, 9));
            this.add(new WoodenFence(0, 8));
            this.add(new WoodenFence(0, 7));
            this.add(new WoodenFence(0, 6));
            this.add(new WoodenFence(0, 5));
            this.add(new WoodenFence(0, 4));
            this.add(new WoodenFence(0, 3));
            this.add(new WoodenFence(0, 2));
            this.add(new WoodenFence(0, 1));

            //Add player.
            //Player moves to mouse click position.
            this.add(new Player(3, 8));

            //Add goal.
            //Go to next level once you reach goal!
            this.add(new Goal(13, 8, Driveway));
        }
}
```

```
class Driveway extends Level
{
	initialize()
	{
		this.setSize(16);

		//Add borders between given points
		this.addBorders([
			[0,0], [0,15], [15,15], [15,0]
		]);

		//Add player.
		this.add(new Player(3, 8));

		//Add goal.
		this.add(new Goal(13, 8, MaenderingRoad));

		//Add obstacles
		this.add(new WoodenFence(8, 4));
		this.add(new WoodenFence(8, 5));
		this.add(new WoodenFence(7, 6));
		this.add(new WoodenFence(7, 7));
		this.add(new WoodenFence(8, 8));
		this.add(new WoodenFence(8, 9));
		this.add(new WoodenFence(7, 10));
		this.add(new WoodenFence(7, 11));
	}
}
```

```
class MaenderingRoad extends Level
{
	initialize()
	{
		this.setSize(16);

		this.addBorders([
			[0,0], [0,15], [15,15], [15,0]
		]);

		this.add(new Player(2, 2));
		this.add(new Goal(14, 14, Alleyway));

		this.add(new WoodenFence(1, 3));
		this.add(new WoodenFence(2, 3));
		this.add(new WoodenFence(3, 3));
		this.add(new WoodenFence(4, 3));
		this.add(new WoodenFence(5, 3));
		this.add(new WoodenFence(6, 3));
		this.add(new WoodenFence(6, 4));
		this.add(new WoodenFence(6, 5));
		this.add(new WoodenFence(6, 6));

		this.add(new WoodenFence(9, 1));
		this.add(new WoodenFence(9, 2));
		this.add(new WoodenFence(9, 3));
		this.add(new WoodenFence(9, 4));
		this.add(new WoodenFence(9, 5));
		this.add(new WoodenFence(9, 6));
		this.add(new WoodenFence(9, 7));
		this.add(new WoodenFence(9, 8));
		this.add(new WoodenFence(9, 9));
		this.add(new WoodenFence(8, 9));
		this.add(new WoodenFence(7, 9));
		this.add(new WoodenFence(6, 9));
		this.add(new WoodenFence(5, 9));
		this.add(new WoodenFence(4, 9));
		this.add(new WoodenFence(3, 9));
		this.add(new WoodenFence(3, 8));
		this.add(new WoodenFence(3, 7));
		this.add(new WoodenFence(3, 6));

		this.add(new WoodenFence(3, 10));
		this.add(new WoodenFence(3, 11));
		this.add(new WoodenFence(3, 12));

		this.add(new WoodenFence(6, 12));
		this.add(new WoodenFence(6, 13));
		this.add(new WoodenFence(6, 14));
```

```
            this.add(new WoodenFence(9, 10));
            this.add(new WoodenFence(9, 11));
            this.add(new WoodenFence(9, 12));

            this.add(new WoodenFence(12, 3));
            this.add(new WoodenFence(12, 4));
            this.add(new WoodenFence(12, 5));
            this.add(new WoodenFence(12, 6));
            this.add(new WoodenFence(12, 7));
            this.add(new WoodenFence(12, 8));
            this.add(new WoodenFence(12, 9));
            this.add(new WoodenFence(12, 10));
            this.add(new WoodenFence(12, 11));
            this.add(new WoodenFence(12, 12));
            this.add(new WoodenFence(12, 13));
            this.add(new WoodenFence(12, 14));
        }
}
```

```
class Alleyway extends Level
{
	initialize()
	{
		this.setSize(32);

		this.addBorders([
			[0,10], [0,21], [31,21], [31,10]
		]);

		this.add(new Player(6, 16));
		this.add(new Goal(26, 16, Paddock));

		//Add hazards.
		//Touching these kills player and level starts from beginning.
		this.add(new ElectricFence(11, 15));
		this.add(new ElectricFence(11, 16));
		this.add(new ElectricFence(11, 17));
		this.add(new ElectricFence(11, 18));
		this.add(new ElectricFence(11, 19));
		this.add(new ElectricFence(11, 20));

		this.add(new ElectricFence(20, 11));
		this.add(new ElectricFence(20, 12));
		this.add(new ElectricFence(20, 13));
		this.add(new ElectricFence(20, 14));
		this.add(new ElectricFence(20, 15));
		this.add(new ElectricFence(20, 16));
	}
}
```

```
class Paddock extends Level
{
	initialize()
	{
		this.setSize(16);

		this.addBorders([
			[0,0], [0,15], [15,15], [15,0]
		]);

		this.add(new Player(2, 8));
		this.add(new Goal(14, 8, Holes));

		//For-loop iterates numbers from 1 to 11
		//and adds hazards
		for(var i = 1; i < 12; i++) {
			this.add(new ElectricFence(3, i));
		}
		this.add(new WoodenFence(3,12));

		for(var i = 4; i < 15; i++) {
			this.add(new ElectricFence(6, i));
		}
		this.add(new WoodenFence(6,3));

		for(var i = 1; i < 12; i++) {
			this.add(new ElectricFence(9, i));
		}
		this.add(new WoodenFence(9,12));

		for(var i = 4; i < 15; i++) {
			this.add(new ElectricFence(12, i));
		}
		this.add(new WoodenFence(12,3));
	}
}
```

```
class Holes extends Level
{
	initialize()
	{
		this.setSize(16);

		this.addBorders([
			[0,0], [0,15], [15,15], [15,0]
		]);

		this.add(new Player(3, 8));
		this.add(new Goal(13, 8, Minefield));

		for(var i = 1; i < 3; i++) {
			this.add(new ElectricFence(5, i));
		}
		for(var i = 5; i < 11; i++) {
			this.add(new ElectricFence(5, i));
		}
		for(var i = 13; i < 15; i++) {
			this.add(new ElectricFence(5, i));
		}
		for(var i = 1; i < 7; i++) {
			this.add(new ElectricFence(10, i));
		}
		for(var i = 9; i < 15; i++) {
			this.add(new ElectricFence(10, i));
		}
	}
}
```

```
class Minefield extends Level
{
	initialize()
	{
		this.setSize(16);

		this.addBorders([
			[0,0], [0,15], [15,15], [15,0]
		]);

		this.add(new Player(3, 8));
		this.add(new Goal(13, 8, River));

		var a = 0;
		for(var x = 2; x < 15; x += 3) {
			for(var y = 2; y < 15; y += 3) {
				//Mines are small circle hazards.
				//They kill people.
				if(a % 2 == 0) this.add(new Mine(x, y));
				else this.add(new HiddenMine(x, y));

				a++
			}
		}
	}
}
```

```
class River extends Level
{
	initialize()
	{
		this.setSize(32);

		this.addBorders([
			[0,0], [0,31], [31,31], [31,0]
		]);

		this.add(new Player(3, 16));
		this.add(new Goal(29, 16, LabLvl2));

		this.mines = [];
		var a = 0;
		for(var i = 2; i < 31; i += 2) {
			if(a % 2 == 0) var mine = new Mine(i, i);
			else var mine = new HiddenMine(i, i);
			a++;

			this.mines.push(mine);
			this.add(mine);
		}
	}

	//Update is run every frame
	update()
	{
		super.update();

		for(var i = 0; i < this.mines.length; i++) {
			this.mines[i].position.y =
				(this.mines[i].position.y + 0.2) % 32;
		}
	}
}
```

```
class LabLvl2 extends Level
{
	initialize()
	{
		this.setSize(64);

		this.addBorders([
			[0,0], [63,0], [63,63], [0,63]
		]);

		this.add(new Player(4, 4));
		this.add(new Goal(36, 28, Home));

		var i;
		//Vertical
		for (i = 1; i < 56; i++) {
			this.add(new WoodenFence(8, i));
		}
		for (i = 16; i < 40; i++) {
			this.add(new WoodenFence(24, i));
		}
		for (i = 24; i < 40; i++) {
			this.add(new WoodenFence(40, i));
		}
		for (i = 8; i < 56; i++) {
			this.add(new WoodenFence(56, i));
		}
		//Horizontal
		for (i = 16; i < 56; i++) {
			this.add(new WoodenFence(i, 8));
		}
		for (i = 32; i < 40; i++) {
			this.add(new WoodenFence(i, 24));
		}
		for (i = 16; i < 48; i++) {
			this.add(new WoodenFence(i, 48));
		}

		//Vertical
		for (i = 8; i < 48; i++) {
			this.add(new ElectricFence(16, i));
		}
		for (i = 24; i < 32; i++) {
			this.add(new ElectricFence(32, i));
		}
		for (i = 16; i < 48; i++) {
			this.add(new ElectricFence(48, i));
		}
		//Horizontal
		for (i = 24; i < 48; i++) {
```

```
                this.add(new ElectricFence(i, 16));
            }
            for (i = 24; i < 40; i++) {
                this.add(new ElectricFence(i, 40));
            }
            for (i = 8; i < 56; i++) {
                this.add(new ElectricFence(i, 56));
            }
        }
}
```

```
class Home extends Level
{
	initialize()
	{
		this.setSize(16);

		this.addBorders([
			[0,0], [0,15], [15,15], [15,0]
		]);

		this.add(new Player(3, 8));
		this.add(new House(13, 8));
	}
}
```