



**Point:** A point in the grid, defined by an x- and a y-coordinate.

**Kangaroo:** Skippy is an instance of this class.

**Grid:** The grid that Skippy is hopping about on, can be of any dimension.

Each class should be put in a file on its own, e.g., the class Kangaroo should be in a file called **Kangaroo.scala**.

Your solution should also include a file called **main.scala** that includes a function with signature `@main def main: Unit` that:

- Instantiates the various classes;
- Starts and runs the simulation;
- Prints the final report, including the die statistics.

Assume the South-West corner of the space is (0,0) and that the grid is a square. Your solution should work for a grid of any size. Assume Skippy starts at (0,0) and that home is the North-East corner of the space at (dimension-1, dimension-1).

### Input/Output Specifications

**Input:** Dimension of the grid. Integer, greater than or equal to 1. Check this on input.

**Output:** A sample program execution is given below (die stats are for example only). Your program should produce output in this format.

### Sample Program Output for Part I

```
Enter dimension of the Grid (>=1):
```

```
100
```

```
Hopped to: (0, 1)
```

```
Oops, hit the boundary: (-1, 1)
```

```
Oops, hit the boundary: (-1, 1)
```

```
Hopped to: (0, 0)
```

```
Oops, hit the boundary: (-1, 0)
```

```
Hopped to: (1, 0)
```

```
...
```

```
...
```

```
Hopped to: (1, 2)
```

```
Hopped to: (0, 2)
```

```
Oops, hit the boundary: (-1, 2)
```

```
Hopped to: (1, 2)
```

```
Hopped to: (2, 2)
```

```
Finished in 18 hops!
```

```
Die statistics:
```

```
Total throws:: 53332
```

```
North: 25.2% South: 24.8% East: 25.2% West: 24.8%
```