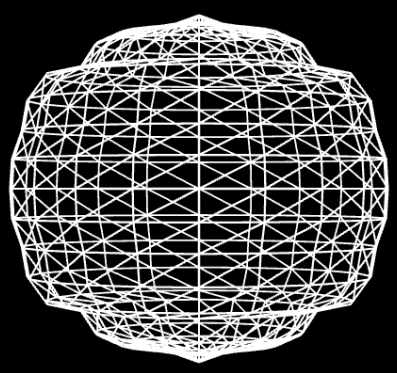
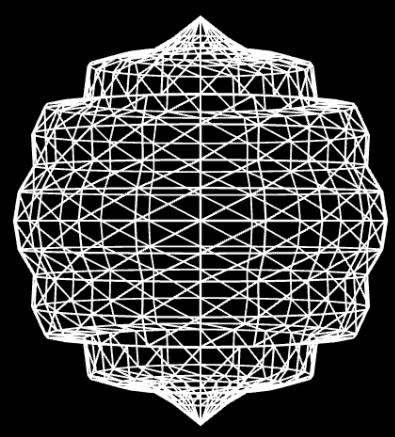
iteration#01 noiseSeed(50)



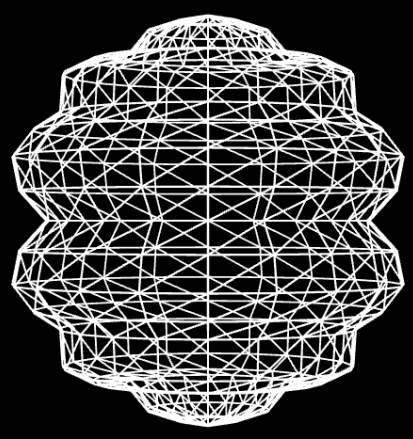
let total = 20;
let radius = 100;
let noiseMax = 01;

iteration#02 noiseSeed(50)



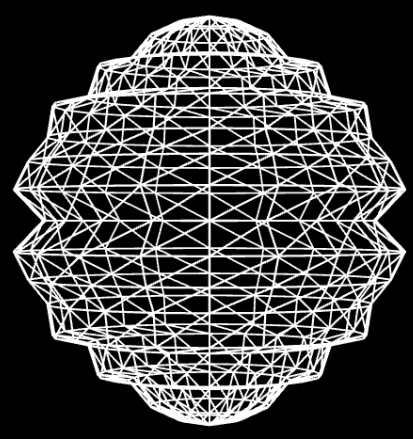
let total = 20;
let radius = 100;
let noiseMax = 02;

iteration#03 noiseSeed(50)



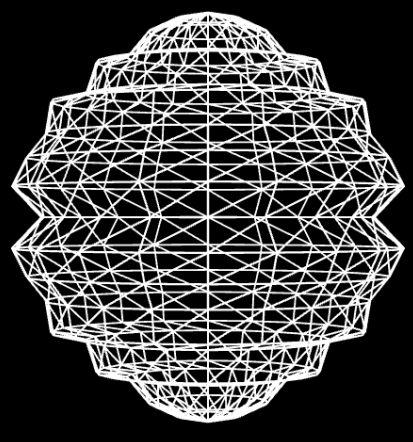
let total = 20;
let radius = 100;
let noiseMax = 03;

iteration#04 noiseSeed(50)



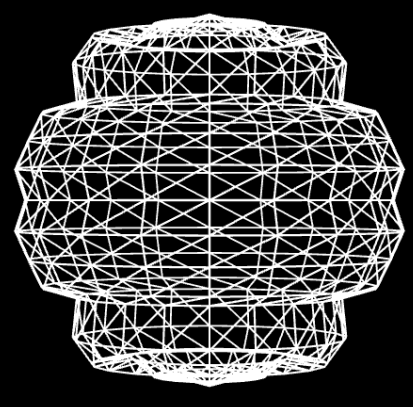
let total = 20;
let radius = 100;
let noiseMax = 04;

iteration#05 noiseSeed(50)



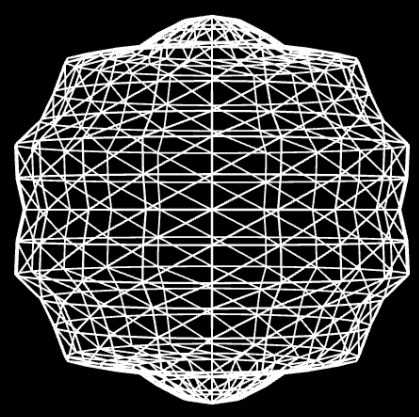
let total = 20;
let radius = 100;
let noiseMax = 05;

iteration#06 noiseSeed(50)



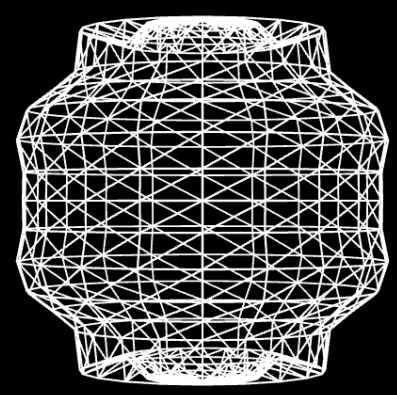
let total = 20;
let radius = 100;
let noiseMax = 06;

iteration#07 noiseSeed(50)



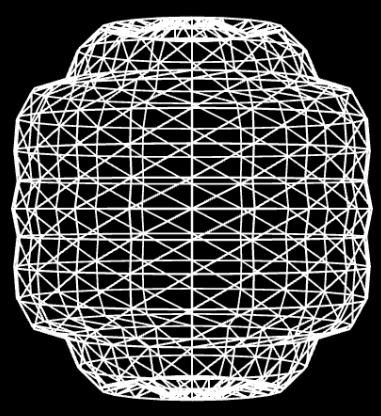
let total = 20;
let radius = 100;
let noiseMax = 07;

iteration#08 noiseSeed(50)



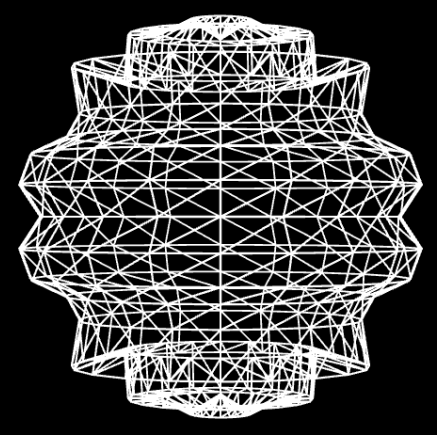
let total = 20;
let radius = 100;
let noiseMax = 08;

iteration#09 noiseSeed(50)



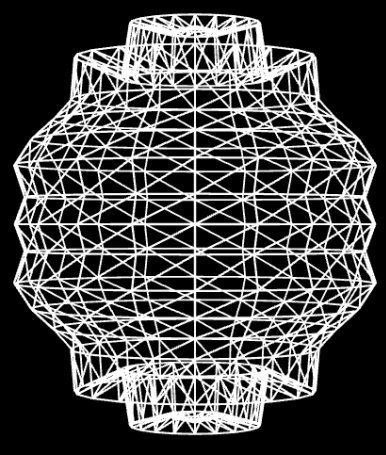
let total = 20;
let radius = 100;
let noiseMax = 09;

iteration#10 noiseSeed(50)



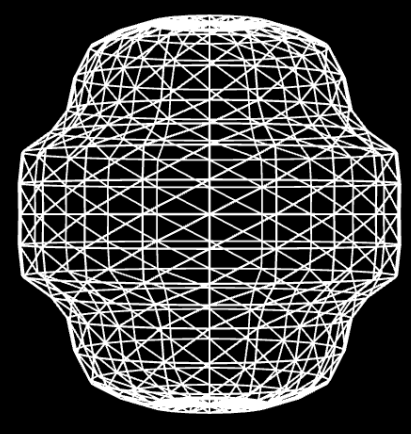
let total = 20;
let radius = 100;
let noiseMax = 10;

iteration#11 noiseSeed(50)



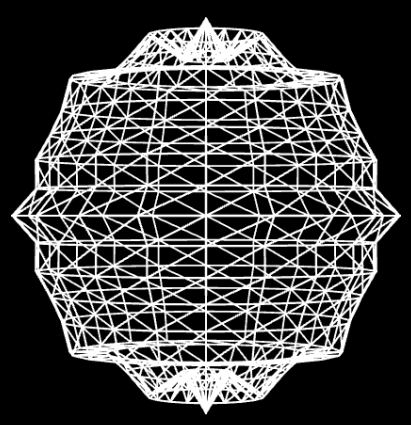
let total = 20;
let radius = 100;
let noiseMax = 11;

iteration#12 noiseSeed(50)



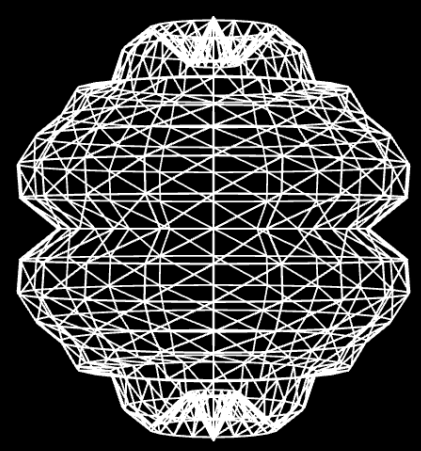
let total = 20;
let radius = 100;
let noiseMax = 12;

iteration#13 noiseSeed(50)



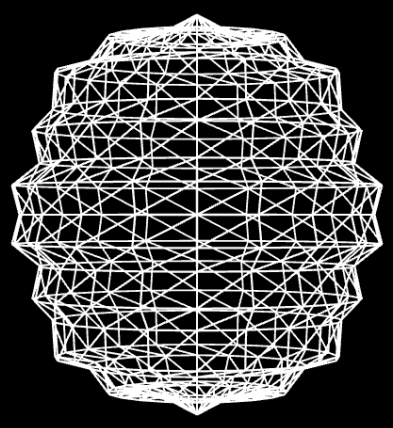
```
let total = 20;
let radius = 100;
let noiseMax = 13;
```

iteration#14 noiseSeed(50)



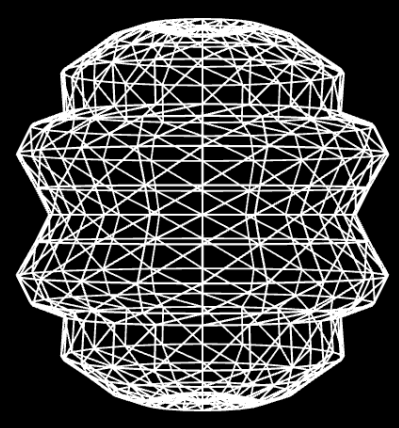
```
let total = 20;
let radius = 100;
let noiseMax = 14;
```

iteration#15 noiseSeed(50)



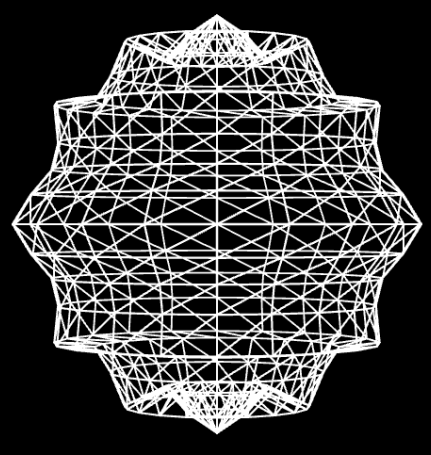
```
let total = 20;
let radius = 100;
let noiseMax = 15;
```

iteration#16 noiseSeed(50)



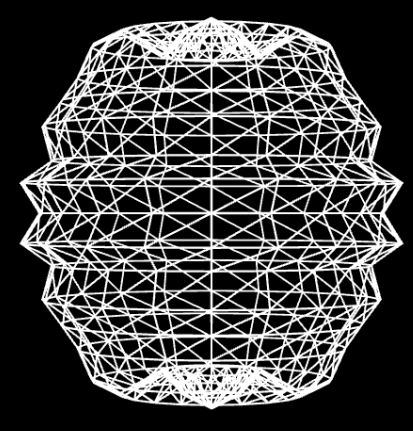
```
let total = 20;
let radius = 100;
let noiseMax = 16;
```

iteration#17 noiseSeed(50)



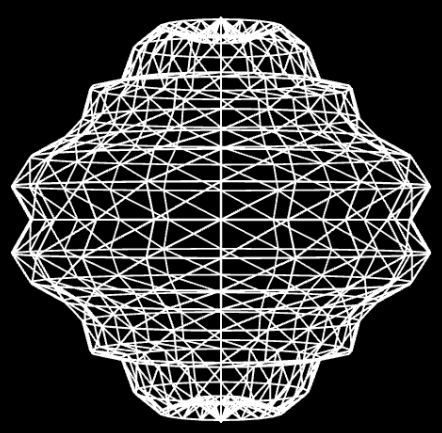
```
let total = 20;
let radius = 100;
let noiseMax = 17;
```

iteration#18 noiseSeed(50)



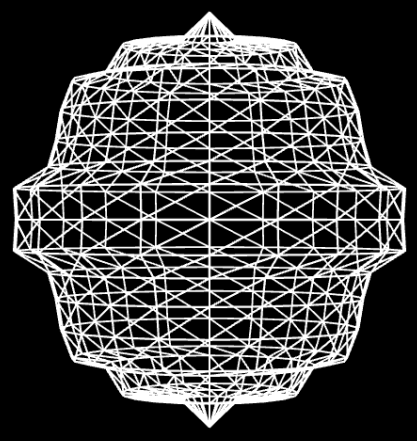
```
let total = 20;
let radius = 100;
let noiseMax = 18;
```

iteration#19 noiseSeed(50)



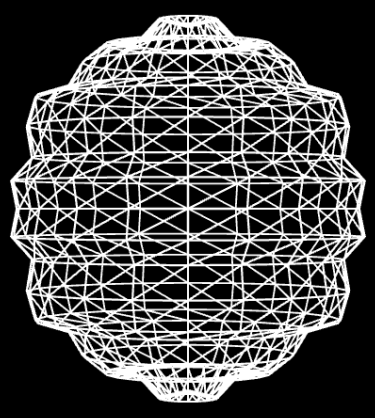
```
let total = 20;
let radius = 100;
let noiseMax = 19;
```

iteration#20 noiseSeed(50)



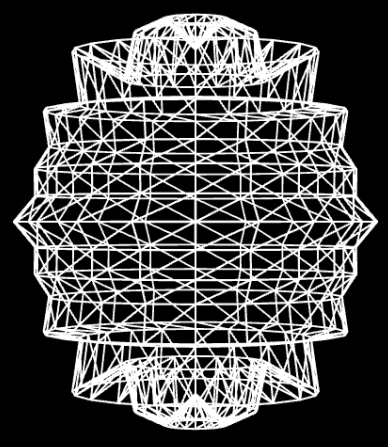
```
let total = 20;
let radius = 100;
let noiseMax = 20;
```

iteration#21 noiseSeed(50)



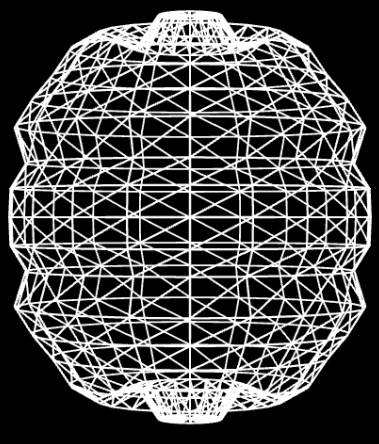
```
let total = 20;
let radius = 100;
let noiseMax = 21;
```

iteration#22 noiseSeed(50)



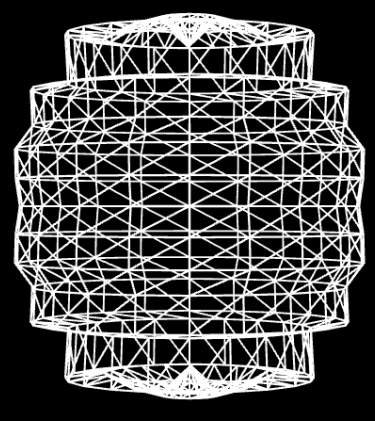
```
let total = 20;
let radius = 100;
let noiseMax = 22;
```

iteration#23 noiseSeed(50)



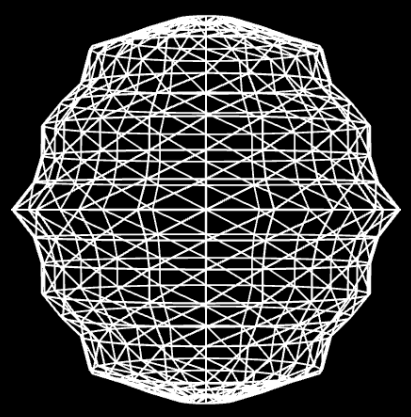
```
let total = 20;
let radius = 100;
let noiseMax = 23;
```

iteration#24 noiseSeed(50)



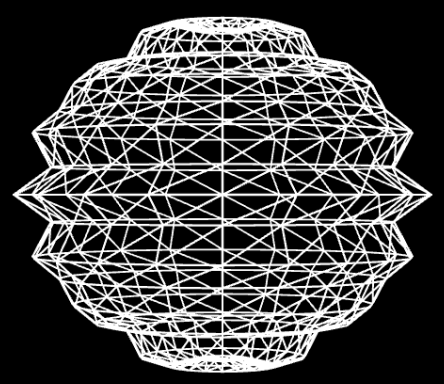
```
let total = 20;
let radius = 100;
let noiseMax = 24;
```

iteration#25 noiseSeed(50)



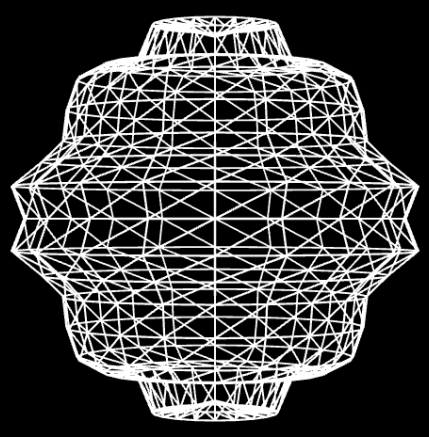
```
let total = 20;
let radius = 100;
let noiseMax = 25;
```

iteration#26 noiseSeed(50)



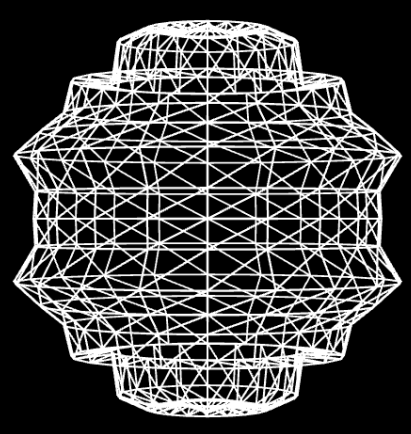
```
let total = 20;
let radius = 100;
let noiseMax = 26;
```

iteration#27 noiseSeed(50)



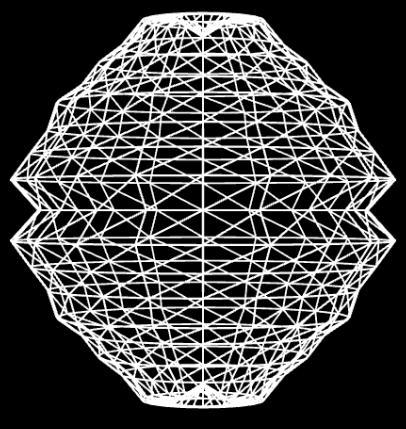
```
let total = 20;
let radius = 100;
let noiseMax = 27;
```

iteration#28 noiseSeed(50)



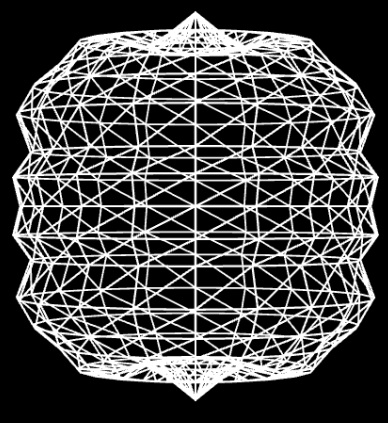
```
let total = 20;
let radius = 100;
let noiseMax = 28;
```

iteration#29 noiseSeed(50)



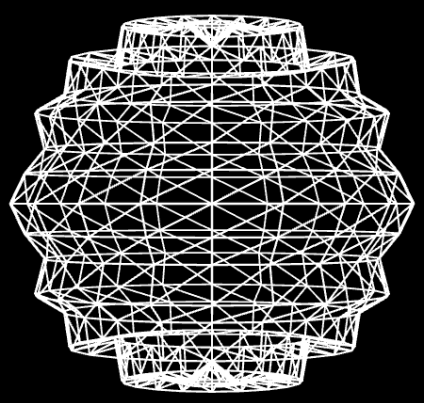
```
let total = 20;
let radius = 100;
let noiseMax = 29;
```

iteration#30 noiseSeed(50)



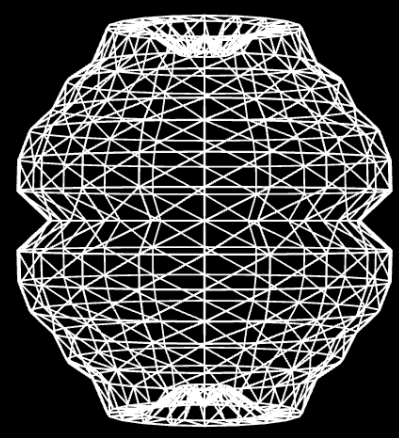
```
let total = 20;
let radius = 100;
let noiseMax = 30;
```

iteration#31 noiseSeed(50)



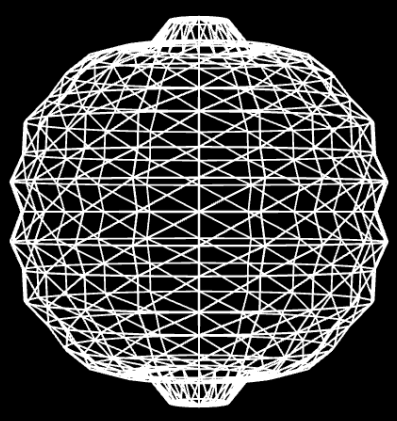
let total = 20;
let radius = 100;
let noiseMax = 31;

iteration#32 noiseSeed(50)



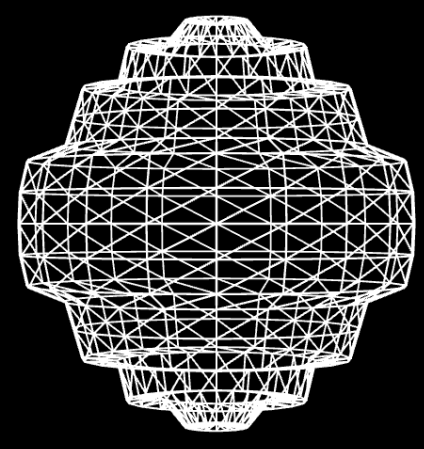
let total = 20;
let radius = 100;
let noiseMax = 32;

iteration#33 noiseSeed(50)



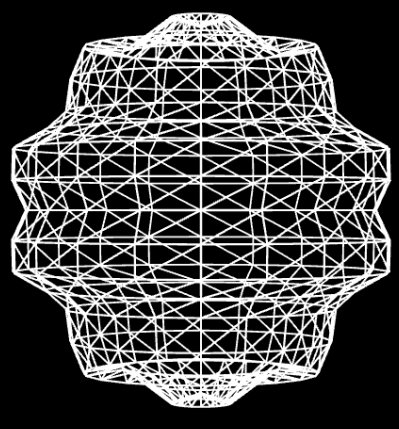
let total = 20;
let radius = 100;
let noiseMax = 33;

iteration#34 noiseSeed(50)



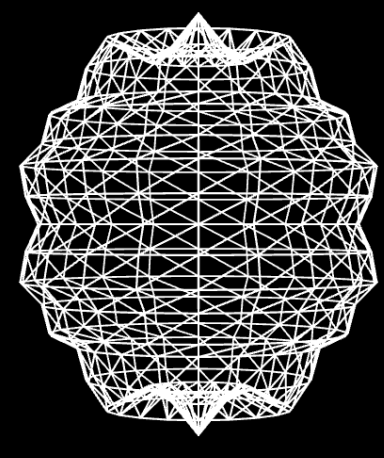
let total = 20;
let radius = 100;
let noiseMax = 34;

iteration#35 noiseSeed(50)



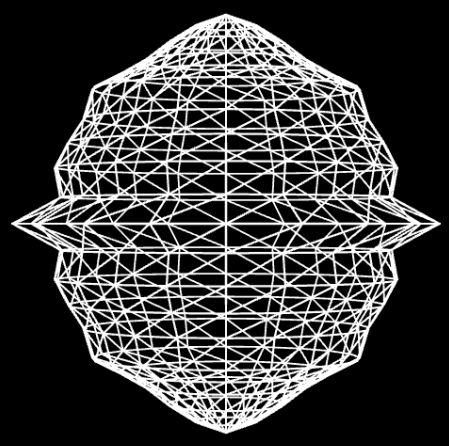
let total = 20;
let radius = 100;
let noiseMax = 35;

iteration#36 noiseSeed(50)



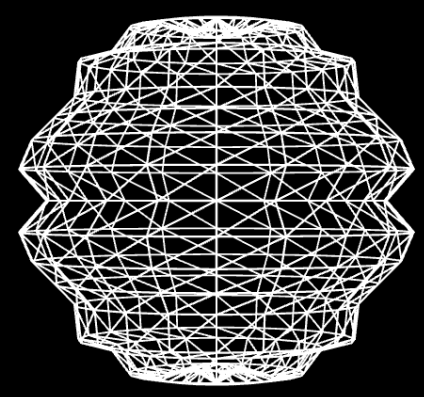
let total = 20;
let radius = 100;
let noiseMax = 36;

iteration#37 noiseSeed(50)



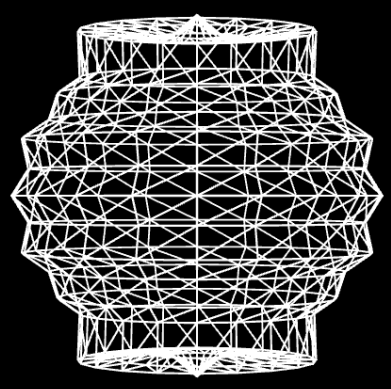
```
let total = 20;
let radius = 100;
let noiseMax = 37;
```

iteration#38 noiseSeed(50)



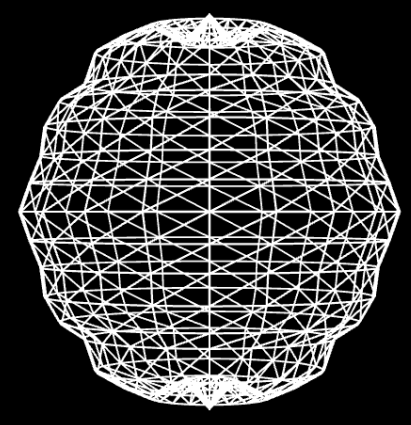
```
let total = 20;
let radius = 100;
let noiseMax = 38;
```

iteration#39 noiseSeed(50)



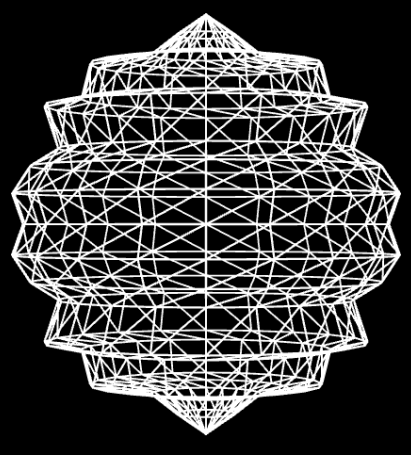
```
let total = 20;
let radius = 100;
let noiseMax = 39;
```

iteration#40 noiseSeed(50)



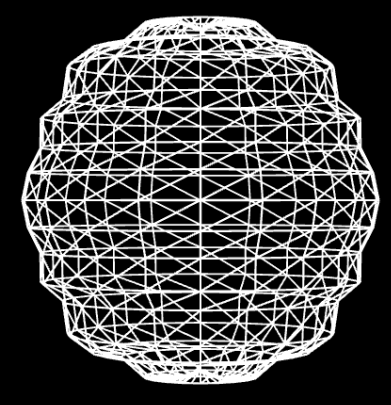
```
let total = 20;
let radius = 100;
let noiseMax = 40;
```

iteration#41 noiseSeed(50)



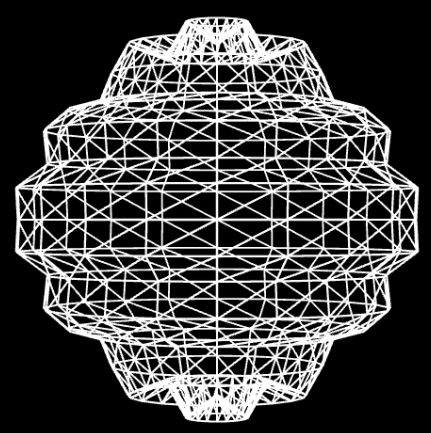
```
let total = 20;
let radius = 100;
let noiseMax = 41;
```

iteration#42 noiseSeed(50)



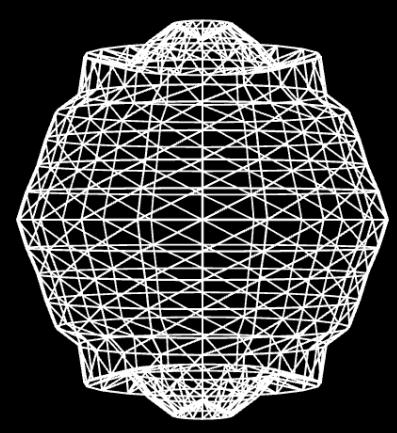
```
let total = 20;
let radius = 100;
let noiseMax = 42;
```

iteration#43 noiseSeed(50)



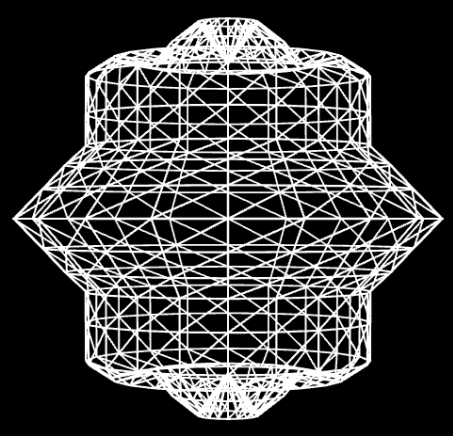
```
let total = 20;
let radius = 100;
let noiseMax = 43;
```

iteration#44 noiseSeed(50)



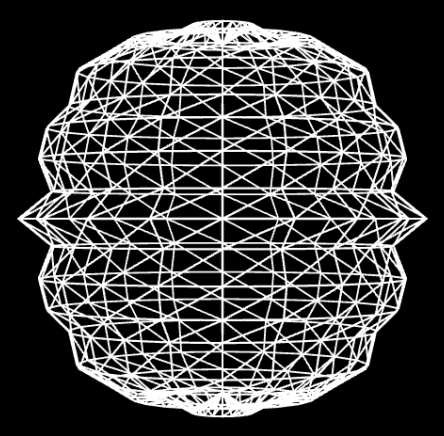
```
let total = 20;
let radius = 100;
let noiseMax = 44;
```

iteration#45 noiseSeed(50)



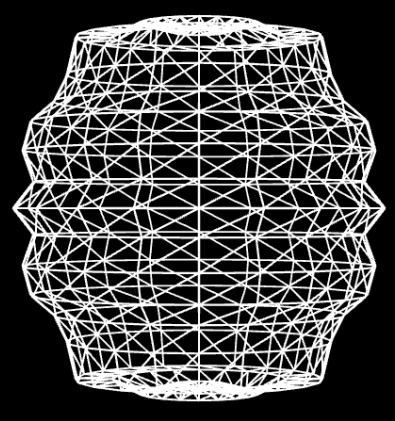
```
let total = 20;
let radius = 100;
let noiseMax = 45;
```

iteration#46 noiseSeed(50)



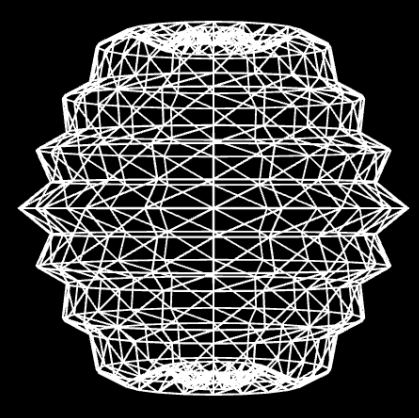
```
let total = 20;
let radius = 100;
let noiseMax = 46;
```

iteration#47 noiseSeed(50)



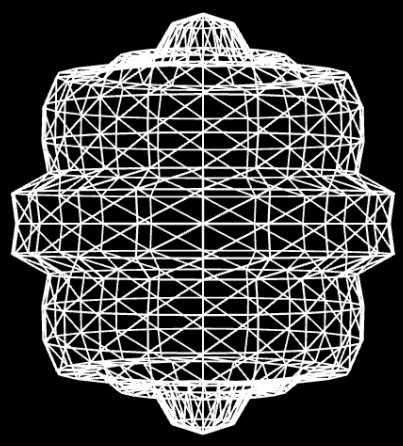
```
let total = 20;
let radius = 100;
let noiseMax = 47;
```

iteration#48 noiseSeed(50)



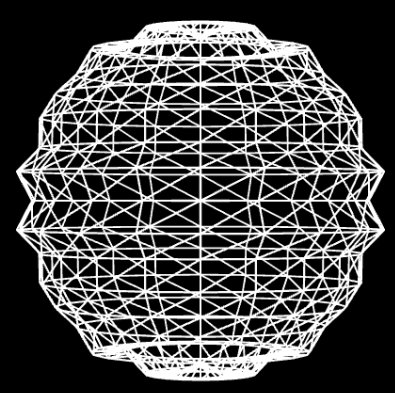
```
let total = 20;
let radius = 100;
let noiseMax = 48;
```

iteration#49 noiseSeed(50)



```
let total = 20;
let radius = 100;
let noiseMax = 49;

function setup() {
  createCanvas(800, 800, WEBGL);
  noFill();
   noiseSeed(50);
}

function draw() {
  background(0);



  let globe = createGlobe(total, radius);
  drawGlobe(globe, total);
}

function createGlobe(total, radius) {
   let globe = Array(total + 1).fill().map(() => Array(total +
1).fill());
  for (let i = 0; i < total + 1; i++) {
    let lat = map(i, 0, total, 0, PI);
    for (let j = 0; j < total + 1; j++) {
      let lon = map(j, 0, total, 0, TWO_PI);
      let noiseVal = noise(cos(lat) * 2 * noiseMax, sin(lat) * 2 *
noiseMax);
      let r = radius + map(noiseVal, 0, 1, -25, 25);
      let x = r * sin(lat) * cos(lon);
      let y = r * cos(lat);
      let z = r * sin(lat) * sin(lon);
      globe[i][j] = createVector(x, y, z);
    }
  }
  return globe;
}

function drawGlobe(globe, total) {
  for (let i = 0; i < total; i++) {
    beginShape(TRIANGLE_STRIP);
    for (let j = 0; j < total + 1; j++) {
      let v1 = globe[i][j];
      let v2 = globe[i + 1][j];
      stroke(255);
      vertex(v1.x, v1.y, v1.z);
      vertex(v2.x, v2.y, v2.z);
    }
    endShape();  }
```

iteration#50 noiseSeed(50)



```
let total = 20;
let radius = 100;
let noiseMax = 50;
```

Harmonic Flux: Exploring Dynamic Geometries Through Parametric Iterations