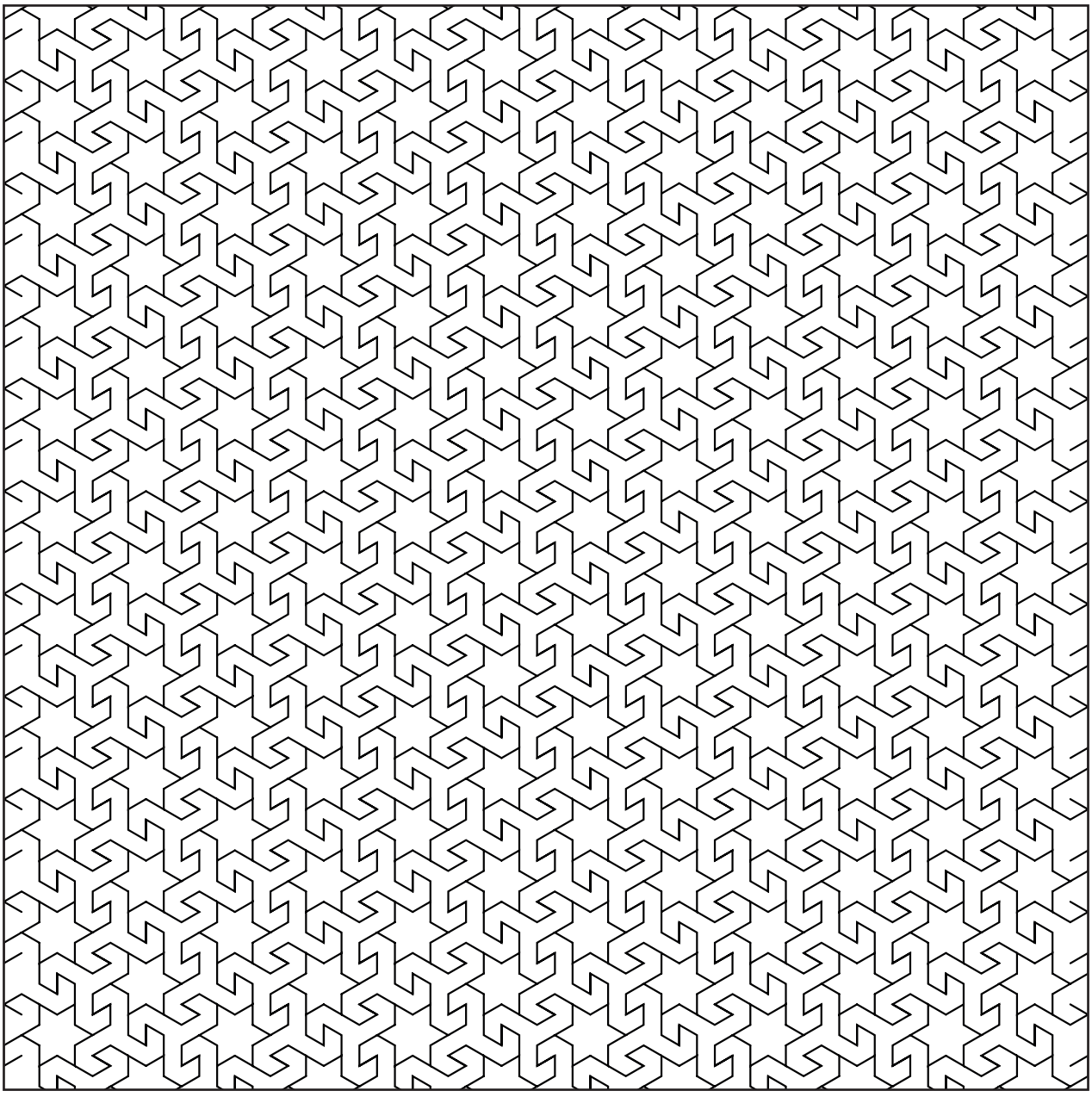
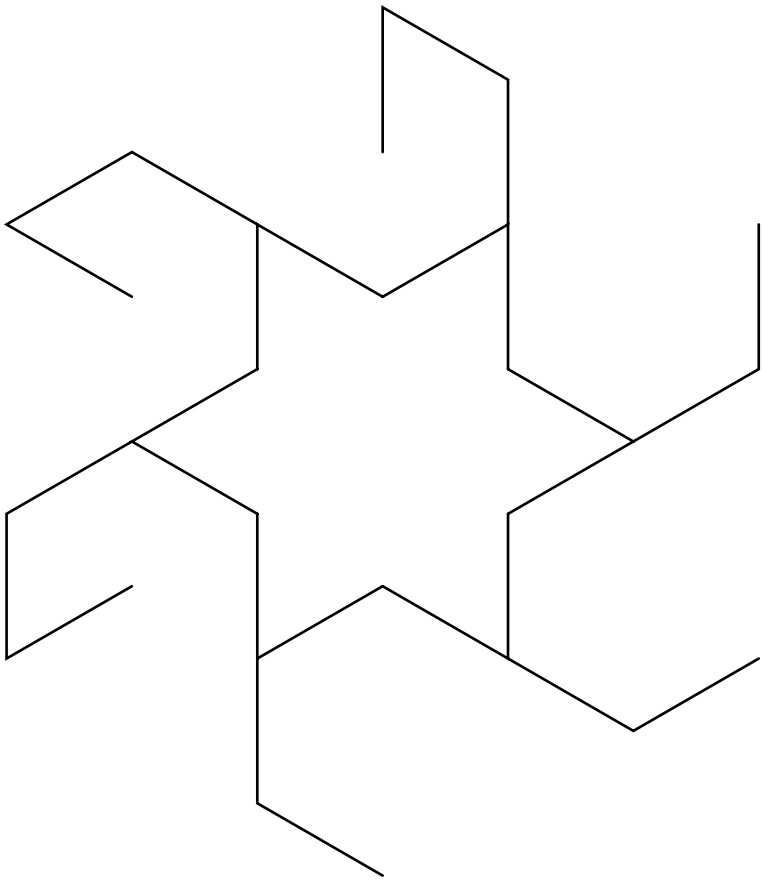
Geometrik Deseni Kodlamak

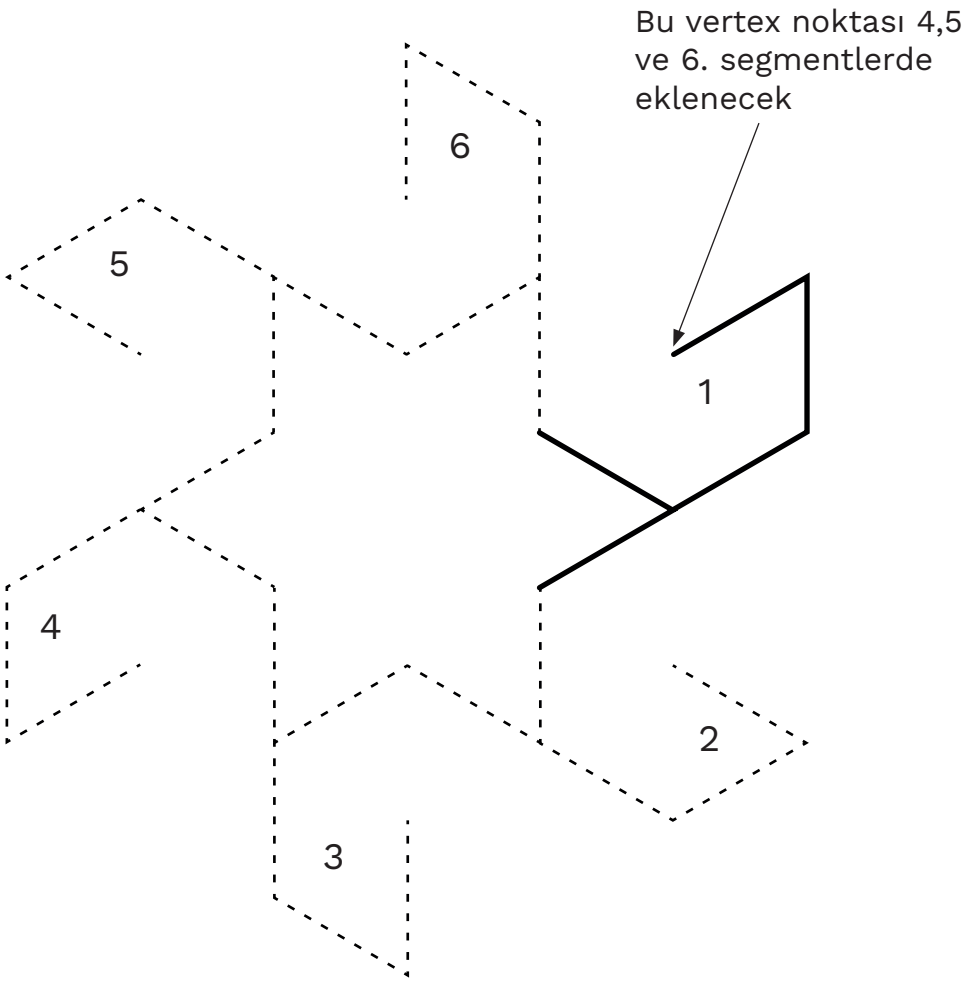
Aşağıdaki deseni inceleyin ve bu deseni oluşturan temel görsel bileşeni bulmaya çalışın.



Motif

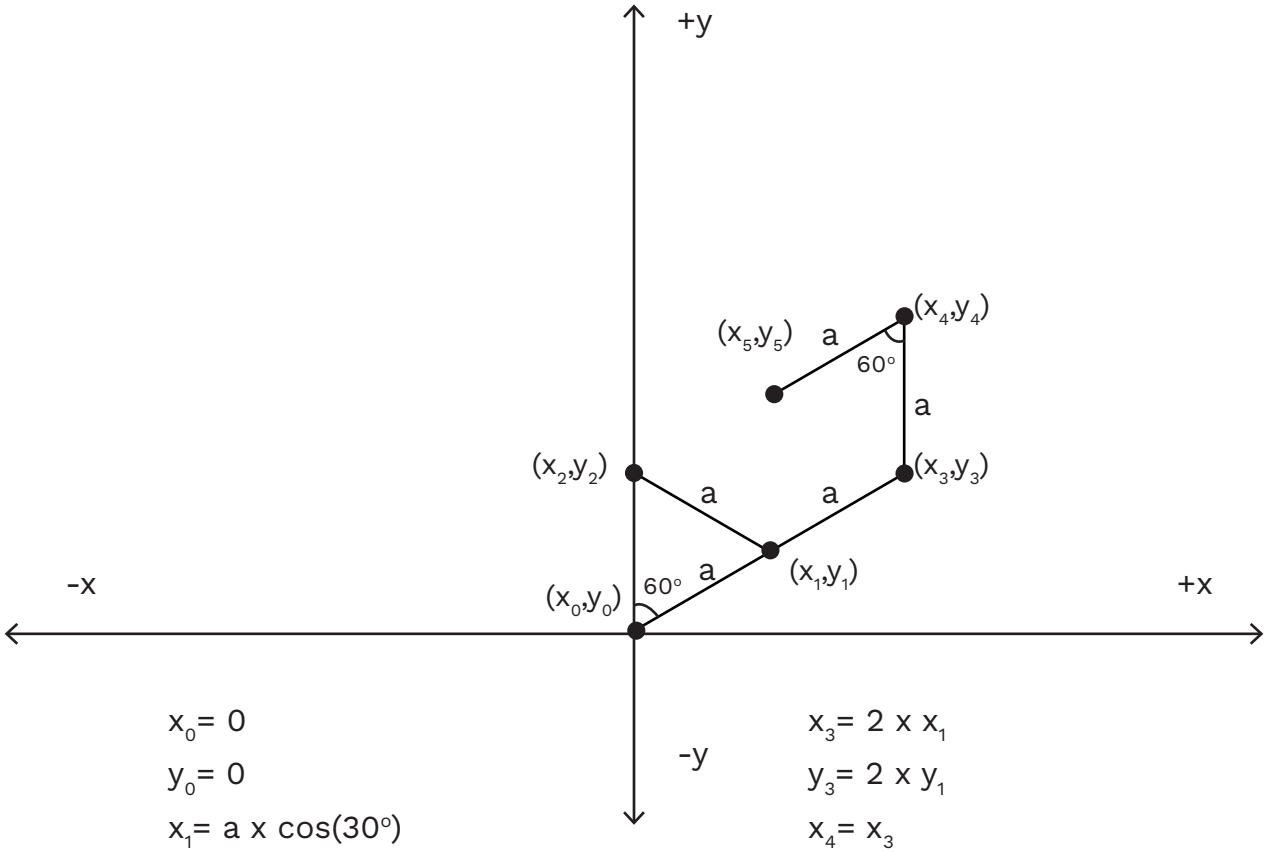


Temel Görsel Bileşini İnceleyelim



Açıları ve Vertex noktalarını tespit etmek

Aşama 1 : Vertex noktalarını bulalım



$x_0 = 0$

$y_0 = 0$

$x_1 = a \times \cos(30^\circ)$

$y_1 = a \times \sin(30^\circ)$

$x_2 = 0$

$y_2 = 2 \times y_1$

$x_3 = 2 \times x_1$

$y_3 = 2 \times y_1$

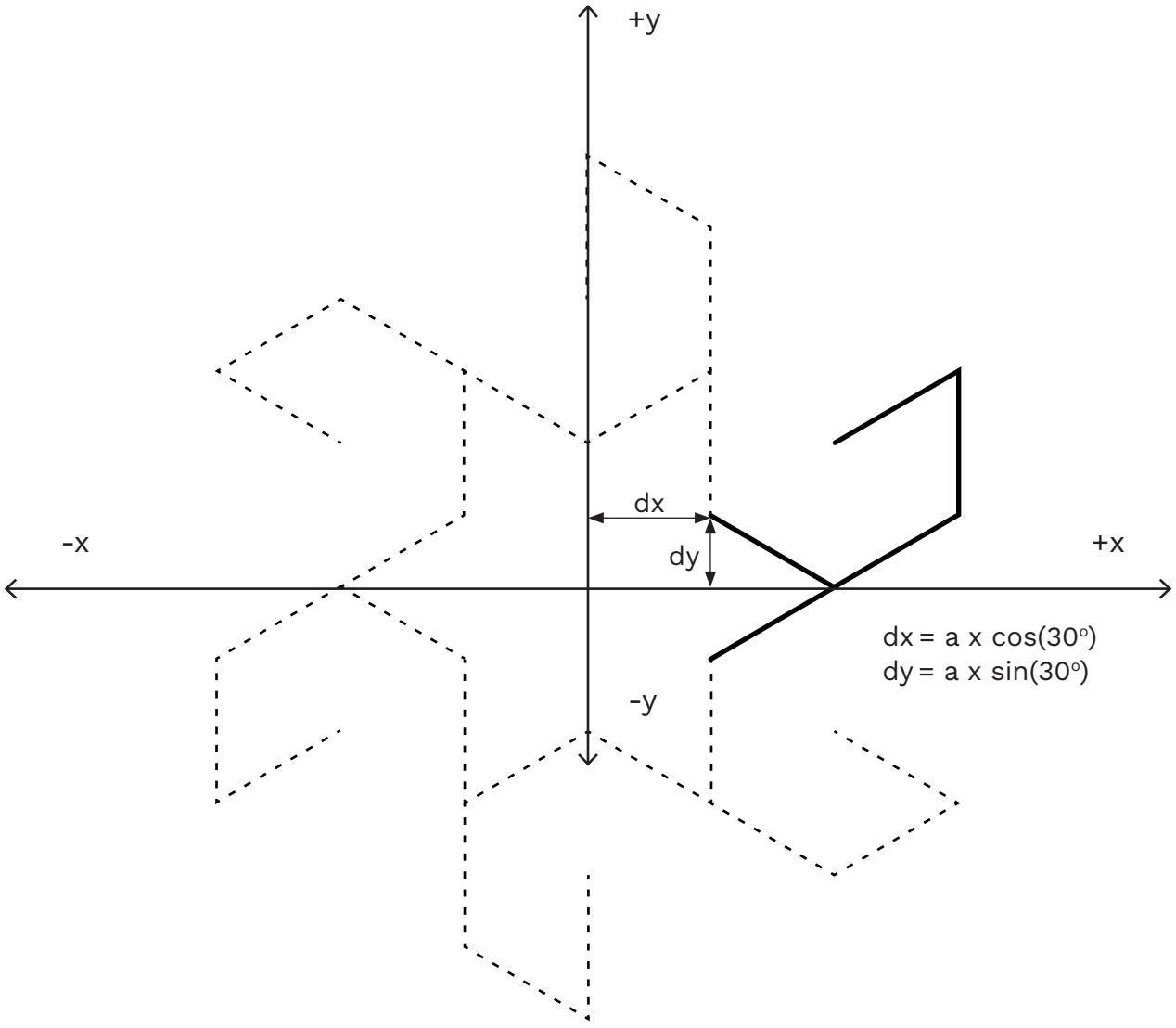
$x_4 = x_3$

$y_4 = y_3 + a$

$x_5 = x_1$

$y_5 = y_1 + a$

Açıları ve Vertex noktalarını tespit etmek



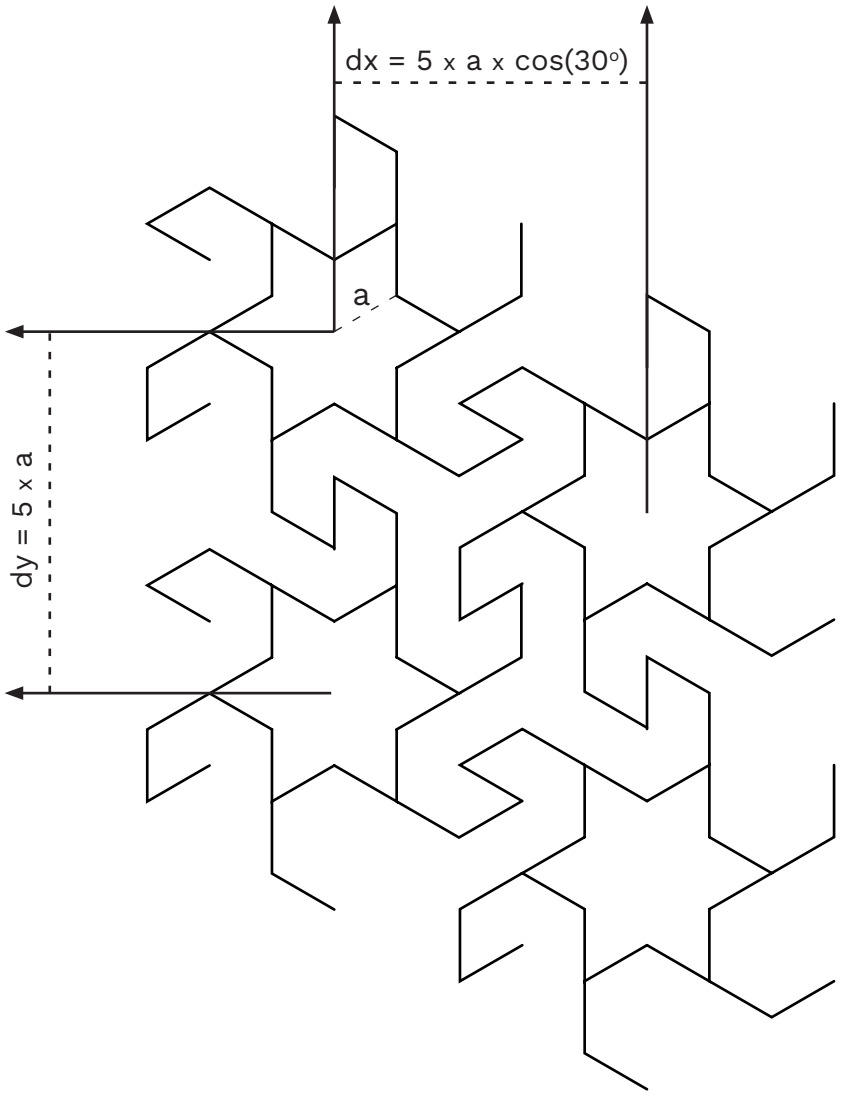
Motifi Oluşturmak

```
/*
Code written by Selcuk ARTUT 2022
Geometric Patterns with Creative Coding
All rights reserved
*/
let a;
function setup() {
  createCanvas(400, 400);
  angleMode(DEGREES);
  noFill();
  a = 40;
  noLoop();
}

function draw() {
  background(255);
  noFill();
  let x0,y0,x1,y1,x2,y2,x3,y3,x4,y4,x5,y5;
  let dx = a * cos(30);
  let dy = a * sin(30);
  push();
  translate(width*0.5,height*0.5);
  for(let i = 0; i<6; i++){
    push();
    rotate(60*i);
    translate(dx,dy);

    beginShape();
    x0 = 0;
    y0 = 0;
    x1 = a * cos(30);
    y1 = a * sin(30);
    x2 = 0;
    y2 = 2 * y1;
    vertex(x0,-y0);
    vertex(x1,-y1);
    vertex(x2,-y2);
    endShape();

    beginShape();
    x3 = 2 * x1;
    y3 = 2 * y1;
    x4 = x3;
    y4 = y3 + a;
    x5 = x1;
    y5 = y1 + a;
    vertex(x1,-y1);
    vertex(x3,-y3);
```

```
    vertex(x4,-y4);
    //special condition to include the 4th,5th and 6th
    if(i>=3){
      vertex(x5,-y5);
    }
    endShape();
    pop();
  }
  pop();
}
```

Bezeme Yapısını İnceleyelim

Aşama 2 : Yukarı ve aşağıya kaymaları belirleyecek xoffset ve yoffset değerlerini hesaplayalım.



Bezeme Kodu

```
/*
Code written by Selcuk ARTUT 2022
Geometric Patterns with Creative Coding
All rights reserved
*/
class Motif {
  constructor(a) {
    this.a = a;
  }

  display() {
    let x0,y0,x1,y1,x2,y2,x3,y3,x4,y4,x5,y5;
    let dx = this.a * cos(30);
    let dy = this.a * sin(30);
    for(let i = 0; i<6; i++){
      push();
      rotate(60*i);
      translate(dx,dy);

      beginShape();
      x0 = 0;
      y0 = 0;
      x1 = this.a * cos(30);
      y1 = this.a * sin(30);
      x2 = 0;
      y2 = 2 * y1;
      vertex(x0,-y0);
      vertex(x1,-y1);
      vertex(x2,-y2);
      endShape();

      beginShape();
      x3 = 2 * x1;
      y3 = 2 * y1;
      x4 = x3;
      y4 = y3 + this.a;
      x5 = x1;
      y5 = y1 + this.a;
      vertex(x1,-y1);
      vertex(x3,-y3);
      vertex(x4,-y4);
      if(i>=3){
        vertex(x5,-y5);
      }
      endShape();
      pop();
    }
  }
}
```

```
let a = 20;
let xOff,yOff;
let nRow;
let nCol;
let motif = new Motif(a);

function setup() {
  createCanvas(800, 800);
  angleMode(DEGREES);
  noFill();
  noLoop();

  xOff = 5 * a * cos(30);
  yOff = 5 * a;
  yOffSet = yOff * 0.5;

  nRow = floor(height / xOff);
  nCol = floor(width / yOff);
}

function draw() {
  push();
  for (let c = 0; c < nCol; c++) {
    for (let r = 0; r < nRow; r++) {
      push();
      translate(xOff * c,  yOff * r);
      if(c%2==1){
        translate(0,  yOffSet);
      }
      motif.display();
      pop();
    }
  }
  pop();
}
```