

## Listingi do lekcji 3: Kodujemy fraktale

### Drzewo binarne

[www.khanacademy.org/cs/drzewobin/2719885296](http://www.khanacademy.org/cs/drzewobin/2719885296)

drzewoBin

```
/**      Drzewo binarne z rekurencją      **/
var alfa = 30; //kąt skrętu (między gałęziami *2)
var skala = 0.5; //zmniejszenie kolejnego poziomu
var dPocz = 150; //długość pnia
var nPoz = 6; //liczba poziomów
stroke (100, 100, 0);
  //grafika żółwia
var np = function(odl) {
  line(0, 0, 0, -odl);
  translate(0, -odl); };
var ws = function(odl) {np(-odl);};
var pw = function(kt) {rotate(kt*PI/180);};
var lw = function(kt) {pw(-kt);};
  //rekurencyjne drzewo binarne
var drzewo = function(n, d, alf) {
  if (n === 0) {return;}
  strokeWeight(n);
  np(d); lw(alf);
  drzewo(n - 1, d * skala, alf);
  pw(2*alf);
  drzewo(n - 1, d * skala, alf);
  lw(alf); ws(d);
};
  //realizacja
resetMatrix();
background(200, 255, 200);
translate(200, 355);
drzewo(nPoz, dPocz, alfa);
```

### Drzewo binarne z losowością

[www.khanacademy.org/cs/drzewobinlos/1014877709](http://www.khanacademy.org/cs/drzewobinlos/1014877709)

DrzewoBinLos

```
/**      Losowe drzewo binarne (jak w Logo)      **/
var alfa = 30; //kąt skrętu (między gałęziami*2)

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego
```

```
var skala = 0.6; //zmniejsz kolejny poziom
var dPocz = 100; //długość pnia
var nPoz = 10; //liczba poziomów
stroke (94, 56, 0);
    //grafika żółwia
var np = function(odl) {
    line(0, 0, 0, -odl);
    translate(0, -odl); };
var ws = function(odl) {np(-odl);};
var pw = function(kt) {rotate(kt*PI/180);};
var lw = function(kt) {pw(-kt);};

var leaf = getImage("cute/GemGreen"); //liść

//rekurencyjne drzewo binarne z losowymi długościami i kątami
var drzewoLos = function(n, d, alf) {
    if (n===0)
        {image(leaf, get, get, 13, 13); return;}
    strokeWeight(n); np(d); lw(alf);
    drzewoLos(n-1, d*skala*random(70, 130)/100, alf*random(70,
130)/100);
    pw(2*alf);
    drzewoLos(n-1, d*skala*random(70, 130)/100, alf*random(70,
130)/100);
    lw(alf); ws(d);};

resetMatrix();
background(200, 255, 255);
translate(200, 375);
drzewoLos(nPoz, dPocz, alfa);
```

### Trójkąt Sierpińskiego

[www.khanacademy.org/cs/trojksierp/2757566516](http://www.khanacademy.org/cs/trojksierp/2757566516)

trojkSierp

```
/**      Trójkąt Sierpińskiego (jak w Logo)      **/
var bok = 360; //największy bok trójkąta
var n = 6; //stopień, liczba zagłębień
stroke(200, 50, 50);
    //grafika żółwia
var np = function(odl) {
    line(0, 0, 0, -odl);
    translate(0, -odl); };
```

```
var ws = function(odl) {np(-odl)};
var pw = function(kt) {rotate(kt*PI/180)};
var lw = function(kt) {pw(-kt)};

//rekurencyjny trójkąt Sierpińskiego
var trojkatSierp = function(a, n) {
  if (n === 0) {return;}
  trojkatSierp(a/2,n-1); np(a); pw(120);
  trojkatSierp(a/2,n-1); np(a); pw(120);
  trojkatSierp(a/2,n-1); np(a); pw(120);
};

//realizacja
background(250, 255, 200);
translate(20, 360); pw(30);
trojkatSierp(bok, n);
```

### Platek Kocha

[www.khanacademy.org/cs/platekkocha/2762339062](http://www.khanacademy.org/cs/platekkocha/2762339062)

PlatekKocha

```
/**          Platek Kocha (jak w Logo)          **/
var bok = 340; //największy bok trójkąta
var n = 4;     //stopień, liczba zagłębień
stroke(50, 50, 200);
//grafika żółwia
var np = function(odl) {
  line(0, 0, 0, -odl);
  translate(0, -odl); };
var ws = function(odl) {np(-odl)};
var pw = function(kt) {rotate(kt*PI/180)};
var lw = function(kt) {pw(-kt)};

//rekurencyjna krzywa Kocha
var kKocha = function(a, n) {
  if (n < 2) {np(a); return;}
  kKocha(a/3,n-1); lw(60);
  kKocha(a/3,n-1); pw(120);
  kKocha(a/3,n-1); lw(60);
  kKocha(a/3,n-1);      };
//realizacja
resetMatrix();
background(200, 255, 255);
translate(28, 298);
```



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI



**WARSZAWSKA  
WYŻSZA SZKOŁA  
INFORMATYKI**

**UNIA EUROPEJSKA  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY**



```
pw(30);  
for(var i = 1; i < 4; i+=1) {  
    kKocha(bok, n); pw(120);}
```