**BARNSLEY GENERATOR**

**LUCRARE PENTRU ATESTAREA COMPETENȚELOR PROFESIONALE**

**Elev:** Onofrei Tudor-Cristian

**Profesor îndrumător:** Șerban Marinel-Paul

**Liceu:** Colegiul Național “Emil Racoviță” Iași

**Sesiunea:** Mai 2019

**Cuprins **



1. De ce am ales această temă de atestat?
2. Ce sunt fractalii? Efectul fluturelui
3. Proprietăți și caracteristici
4. Metode de generare
5. Fractalii și știința. Fractali din natură
6. Fractalul lui Barnsley
7. Pagina web “Barnsley Generator”
8. Imagini generate de aplicația din interiorul paginii web
9. Partea de cod
10. Posibilități de îmbunătățire
11. Resurse
12. Bibliografie

** De ce am ales această temă de atestat? **

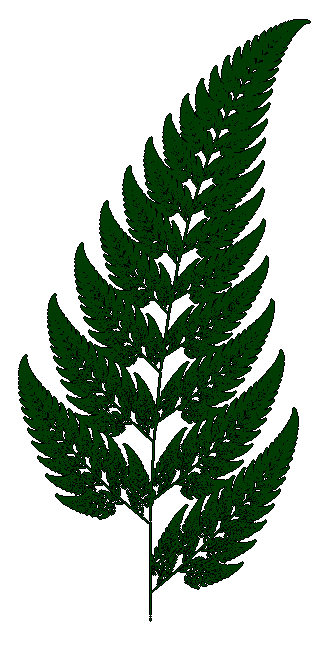
De când eram mic, matematica a reprezentat, pentru mine, unul dintre domeniile de studiu preferate. Odată ajuns la liceu și făcând cunoștință cu informatica despre care se spunea la începutul clasei a-IX-a că “nu o poți învăța fără să știi matematică”, am început să caut intersecția celor 2 domenii. Și am descoperit că este foarte vastă, informatica fiind cea care face posibile multe lucruri inaccesibile unui matematician ce ar lucra exclusiv pe o foaie de hârtie.

Proiectul “Barnsley Generator” este suma pasiunii pentru matematică, pentru informatică și a unei provocări personale: cea de a aprofunda lucrul cu pagini web de care mă temeam încă din clasa a-IX-a. El pentru început a reprezentat un simplu proiect de teză pe care am decis să-l îmbunătățesc din cauza faptului că nu-mi ieșise cum am plănuit la început și de a-l face mult mai atractiv.

Încă din punctul în care am scris prima linie de cod eram conștient de faptul că ceea ce am început e ceva dificil, complicat și anevoios; nu am mai lucrat niciodată în afara orelor de informatică folosind PHP, JavaScript, CSS sau HTML. Pentru a-mi fi mai ușor, m-am transformat într-un traducător: am început prin a scrie unele programe pe care urma să le includ în pagina web în limbajul C++ și mai apoi am continuat a le scrie în limbajele de programare compatibile cu paginile web.

Acum că pagina web e finalizată, sunt mulțumit de faptul că am reușit să-mi depășesc o limită. Am reușit să realizez ceva nou, să lucrez cu o tehnologie pe care nu o stăpâneam deloc până acum și, în același timp, sunt mulțumit că am reușit să creez o aplicație prin care mai multe persoane își pot lăsa amprenta asupra a ceea ce generează pagina web.

**Ce sunt fractalii?** 

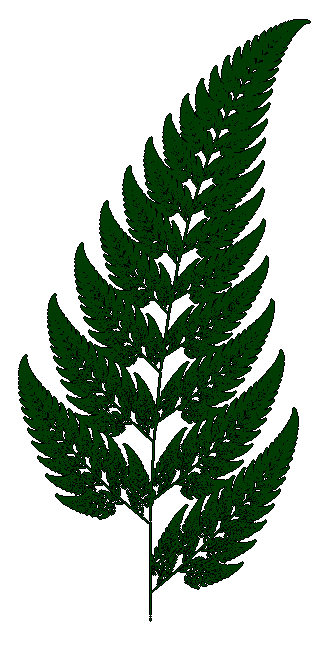


Termenul a fost introdus în 1975 de Benoît Mandelbrot și provine din latinescul “fractuus” care înseamnă “fragmentat” sau “fracturat”.

Colocvial, un fractal este o figură geometrică fragmentată sau frântă care poate fi divizată în mai multe părți astfel încât fiecare dintre acestea să fie (cel puțin aproximativ) o copie miniaturală a întregului.

Teoretic, fractalii sunt infinit similari cu ei înșiși, neputând fi limitați la modele geometrice, dar putând descrie procese de-a lungul timpului. Modele de fractali având un anumit grad de similaritate cu ei înșiși au fost reprezentați sau studiați pe scară largă în imagini, structuri și sunete și au fost identificați în diverse domenii: natură, tehnologie, artă, arhitectură și drept.

**Efectul fluturelui**  



Enunțat de Edward Lorenz în 1945 ca fiind o parte din “Teoria Haosului”, el era descris astfel:



*“Mișcarea aripilor unui fluture azi poate produce o mică schimbare a atmosferei. Din această cauză și de-a lungul unei anumite perioade de timp, atmosfera se va schimba. Peste o lună poate, o tornadă care trebuia să lovească coasta Indoneziei nu va mai apărea. Sau din contră, tocmai din această cauză va apărea.”*

În domeniul teoriei haosului, fractalii au o relevanță particulară din moment ce graficele celor mai haotice procese sunt fractali.

Efectul fluturelui subliniază dependența generării unui fractal de condițiile inițiale; cea mai mică schimbare a condițiilor inițiale duce la rezultate complet diferite.

* **Proprietăți și caracteristici** 

Fractalul, ca obiect geometric, are următoarele caracteristici:

1. Are o structură fină la scări arbitrar de mici.
2. Este prea neregulat pentru a fi descris în limbaj geometric euclidian tradițional.
3. Este autosimilar (măcar aproximativ).
4. Are o definiție simplă și recursivă.

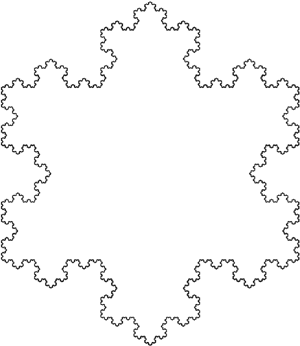
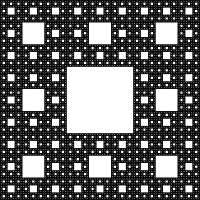
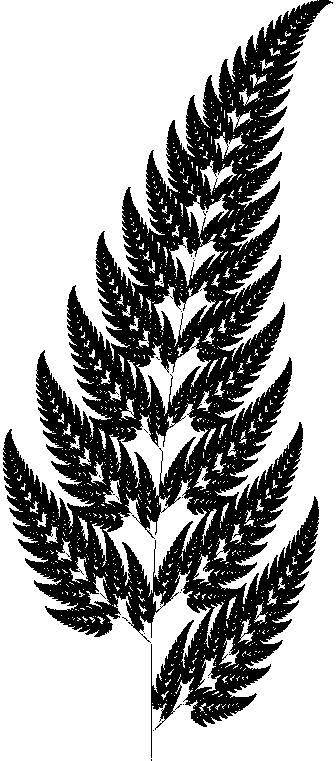
* **Metode de generare** 

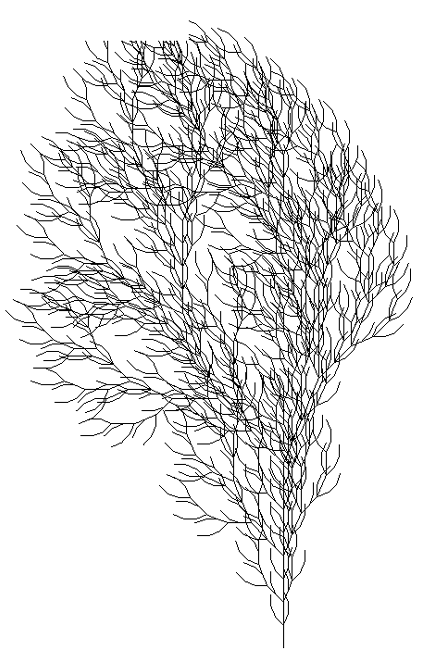
Imagini ai fractalilor pot fi create folosind programe de generare ai fractalilor. Cele mai comune tehnici de generare ai fractalilor sunt:

1. IFS (sistem de funcții iterate): folosește reguli de geometrie fixă pentru înlocuire

Exemple:

“fulgul lui Koch” “Covorul lui Sierpinsky” “feriga lui Barnsley”



1. Sistem Lindenmayer: rescrierea unui șir de caractere

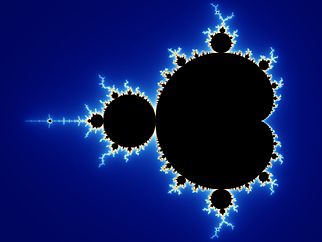
Exemplu:

Următorul copac este generat pe baza șirului definit astfel:

Primul termen al șirului este F. La fiecare generație, F devine ”FF+[+F-F-F]-[-F+F+F]”.

Fiecare caracter are o semnificație: pentru “F” vom desena o linie dreaptă, pentru “+” vom înclina modul de desen cu 30°, pentru “-” vom înclina modul de desen cu -30°, pentru “[” vom pune pe stivă următoarea poziție din șir, iar pentru “]” o vom elimina.

Copacul va arăta în felul următor:

1. Regulă de recurență – cunoscuți și ca fractali “orbită”

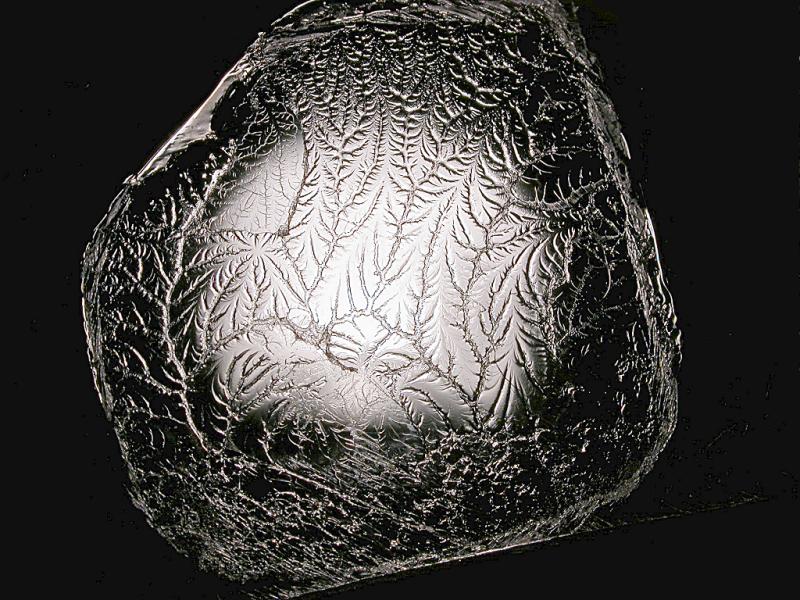
Exemplu:

Fractalul Mandelbrot se definește ca fiind totalitatea punctelor pentru care, aplicând în mod repetat polinomul complex z2+c (pornind de la z = 0 și c fiind un număr cunoscut) rezultatul rămâne în interiorul unui disc de rază finită.

* **Fractalii și știința. Fractali din natură** 

Aplicabilitatea geometriei fractale nu se rezumă doar la fenomene statice, ci și în studiul fenomenelor dinamice, în evoluție.

Despărțirea a două plăci de acril lipite Degradarea unui bloc de 4 țoli sub acțiunea unui curent de înaltă tensiune



Fractalii aproximativi pot fi observați ușor în natură. Aceste obiecte afișează o structură auto-similară la o scare mare, dar finită. Exemplele includ norii, fulgii de zăpadă, cristalele, lanțurile montane, fulgerele, rețelele de râuri, conopida sau broccoli și sistemul de vase sanguine sau pulmonare.

Ferigile și arborii sunt fractali naturali și pot fi modelați ușor pe calculator folosind un algoritm recursiv.

În imaginea alăturată avem un Broccoli Romanesco.

* **Fractalul lui Barnsley** 



Fractalul lui Barnsley sau “Feriga lui Barnsley” este un fractal care a fost denumit după matematicianul britanic Michael Barnsley pe care l-a descris prima dată în cartea sa, “Fractali pretutindeni”. El a făcut ca acest fractal să semene după o ferigă comună, *Asplenium adiantum-nigrum.*

Generarea fractalului:

Generarea sa se face pe calculator, folosind un sistem de funcții iterate.

Având la dispoziție matricea de valori a lui Barnsley, pe care o vom denumi C, și primul punct generat O(0,0), originea sistemului, fractalul este generat alegând în mod aleatoriu un număr de la 1 la 100, aplicând în funcție de valoarea sa una dintre cele 4 funcții pentru a obține coordonatele următorului punct de desenat, desenând punctul respectiv și aplicând același algoritm pentru un alt număr ales în mod aleator și modificând ultimele coordonate găsite.

Matricea C:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0.16 | 0 | 0 |
| 0.85 | 0.04 | -0.04 | 0.85 | 0 | 1.6 |
| 0.20 | -0.26 | 0.23 | 0.22 | 0 | 1.6 |
| -0.15 | 0.28 | 0.26 | 0.24 | 0 | 0.44 |

Cele 4 legi prin care este generat fractalul sunt:

Fi : xn+1 = C[i][1]\* xn + C[i][2] \* yn + C[i][5]; unde i ia valori de la 1 la 4.

yn+1 = C[i][3]\* xn + C[i][4] \* yn + C[i][6];

Fi este aplicată dacă numărul generat aleatoriu este mai mic sau egal cu , unde pi este un factor de probabilitate: p1 = 1; p2 = 85; p3 = p4 = 7.

Modificând valorile variabilelor C și schimbând factorii de probabilitate vom obține specii diferite de ferigi, aici fiind evidențiat “Efectul fluturelui”.

**Pagina web “Barnsley Generator” **



****

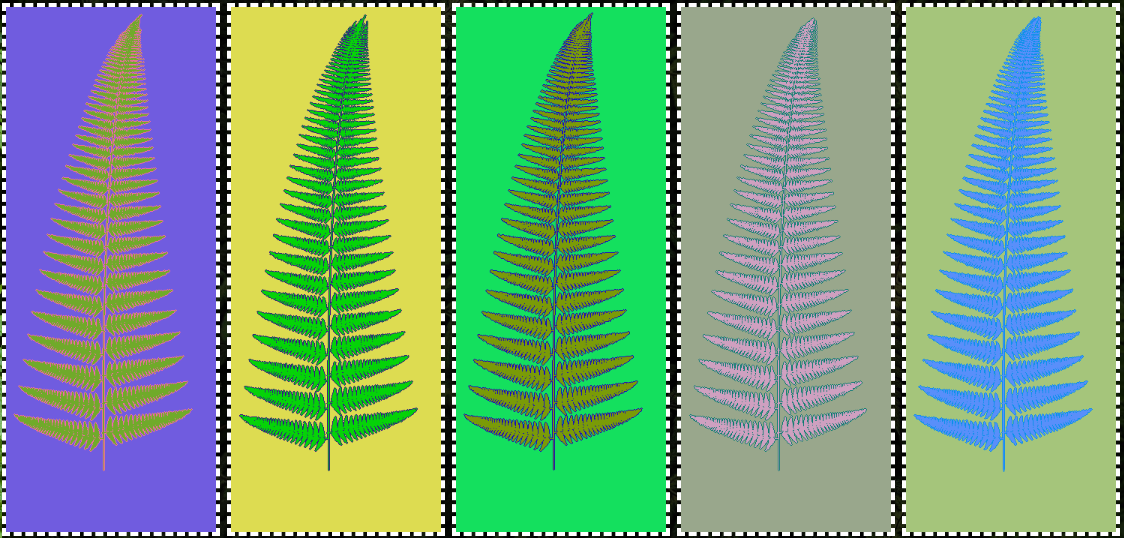
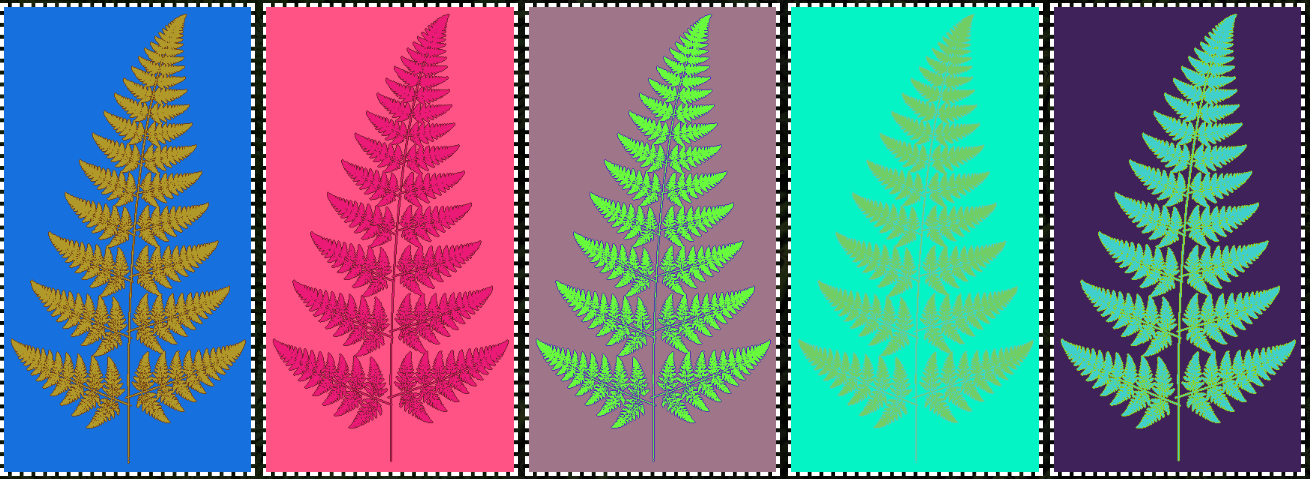
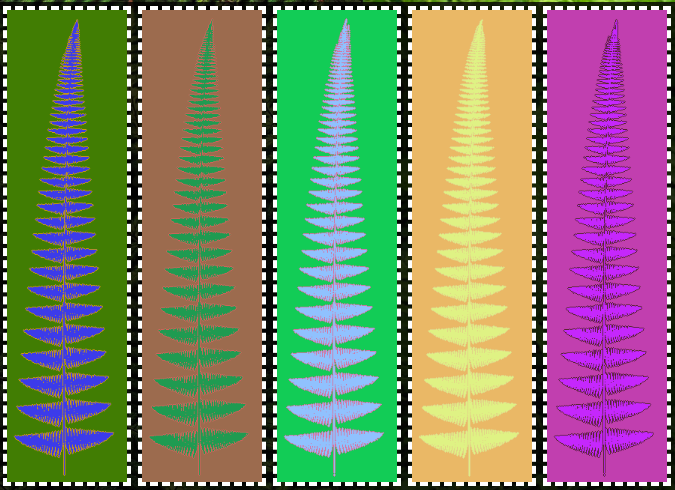
Pagina web “Barnsley Generator” nu face altceva decât să deseneze diferiți fractali Barnsley în funcție de parametrii aleși de noi.

Parametrii sunt:

1. tipul fractalului: fractalul Barnsley are 4 variante principale în care poate apărea, fiecare tip fiind diferit unul față de celălalt
2. culoarea pe care o va avea viitorul fractal
3. culoarea umbrei pe care o va avea fractalul
4. culoarea fundalului pe care va fi desenat fractalul
5. numărul de puncte pe care-l va avea fractalul

Ea pune la dispoziție și o galerie unde putem vizualiza toți fractalii pe care i-am generat până acum.

 **Imagini generate de aplicația din interiorul paginii web **

1. Modelul Barnsley Clasic
2. Modelul Cyclosorus
3. Modelul Culcita Dubia
4. Modelul Fishbone

 **Partea de cod **

Fișierul *Index.html*:

<!DOCTYPE html><!-- declaram fisierul de tip HTML -->

<html lang=**"en"**><!-- setam limba de scriere a documentului HTML -->

<meta charset=**"UTF-8"**><!-- setam modul de codificare al caracterelor -->

<title> **Barnsley Generator** </title><!-- setam titlul barei din browser -->

<link rel=**"icon"** href=**"Imagini/Icon2.png"**><!-- setam iconita din dreptul barei -->

<link rel=**"stylesheet"** href=**"Design1.css"**><!-- legam printr-un link fisierul CSS "Design1.css" pentru a putea accesa diferitele taguri formatate -->

<body><!-- deschidem corpul paginii -->

<p class=**"central"**><img src=**"Imagini/Imagine\_Index4.png"** alt=**""**></p><!-- desenam intr-un paragraf la centru imaginea "Barnsley Generator" -->

<form action=**"Creeaza\_Fractal.php"** method=**"post"**><!-- cream un formular pentru introducerea datelor referitoare la fractal -->

<p> **Alege tipul fractalului:** <!-- afisam intr-un paragraf totul -->

<select name=**"tip"**><!-- cream un tag de tip "select" in care specificam optiunile pentru tipul de fractal -->

<option value=**"Clasic"**> **Clasic** </option>

<option value=**"Cyclosorus"**> **Cyclosorus** </option>

<option value=**"Culcita dubia"**> **Culcita dubia** </option>

<option value=**"Fishbone"**> **Fishbone** </option>

</select><!-- inchidem tagul "select" -->

</p><!-- incheiem paragraful -->

<div class=**"centru"** style=**"font-family:Times New Roman"**> **Seteaza culorile fractalului, a umbrei si a fundalului:** <input name=**"culoare\_frunza"** type=**"color"** value=**"#ff0000"**><input name=**"culoare\_umbra"** type=**"color"** value=**"#ff0000"**><input name=**"culoare\_fundal"** type=**"color"** value=**"#ff0000"**></div><!-- cream cele 3 butoane prin care introducem culorile pentru fractalul ce va urma a fi creat -->

<p> **Alege numarul de puncte din care sa fie format fractalul:** <!-- afisam intr-un paragraf totul -->

<select name=**"calitate"**><!-- cream un tag de tip "select" in care specificam optiunile pentru numarul de puncte din care sa fie format fractalul -->

<option value=**"Simplu"**> **400000** </option>

<option value=**"Mediu"**> **700000** </option>

<option value=**"Complex"**> **1000000** </option>

</select><!-- inchidem tagul "select" -->

</p><!-- incheiem paragraful -->

<div class=**"centru"**><input name=**"Genereaza"** type=**"submit"** value=**"Genereaza fractal"**><input name=**"Random\_colors"** type=**"submit"** value=**"Genereaza fractal de culori random"**><input name=**"Random\_fractal"** type=**"submit"** value=**"Genereaza fractal complet random"**><input name=**"Galerie"** type=**"submit"** value=**"Galerie"**></div><!-- Cream butoanele pentru optiunile de alegere al datelor de creare pentru fractal: crearea fractalului - folosind datele introduse, - de culori aleatorii, - complet aleatoriu sau butonul pentru accesare al galeriei -->

</form><!-- inchidem formularul -->

</body><!-- inchidem corpul paginii -->

</html><!-- inchidem tagul HTML al paginii -->

Fișierul *Creeaza\_Fractal.php*:

<?php

**session\_start**(); // incepem o sesiune PHP pentru salvarea unor valori

?>

<!DOCTYPE html><!-- declaram o pagina HTML -->

<html lang=**"en"**><!-- setam limba de scriere a documentului HTML -->

<meta charset=**"UTF-8"**><!-- setam modul de codificare al caracterelor -->

<title> **Barnsley Generator** </title><!-- setam titlul barei din browser -->

<link rel=**"icon"** href=**"Imagini/Icon2.png"**><!-- setam iconita din dreptul barei -->

<link rel=**"stylesheet"** href=**"Design2.css"**><!-- legam printr-un link fisierul CSS "Design2.css" pentru a putea accesa diferitele taguri formatate -->

<link rel=**"stylesheet"** href=**"Design1.css"**><!-- legam printr-un link fisierul CSS "Design1.css" pentru a putea accesa diferitele taguri formatate -->

<body><!-- deschidem corpul paginii -->

<?php // deschidem o sursa PHP

$directoare = **array**("Clasic", "Cyclosorus", "Culcita dubia", "Fishbone"); // declaram un vector cu numele fisierelor text din care vom citi coeficientii necesari generarii unui tip de fractal Barnsley

$numar\_puncte = **array**(400000, 700000, 1000000); // declaram un vector cu numerele de puncte pe care-l va contine fractalul de generat

$dl = **array**(-1, 0, 1, 0); // declaram vectorii de directie pentru cele 4 puncte cardinale: N S E V

$dc = **array**(0, 1, 0, -1);

$px = $py = 0;

**function** random\_hex\_color(){ // functia returneaza aleatoriu codul in baza 16 a unei culori

$rand = **array**('0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'A', 'B', 'C', 'D', 'E', 'F'); // codul este format dintr-un '#' + 6 caractere in baza 16 scrise unul dupa altul

$color = '#'.$rand[**rand**(0, 15)].$rand[**rand**(0, 15)].$rand[**rand**(0, 15)].$rand[**rand**(0, 15)].$rand[**rand**(0, 15)].$rand[**rand**(0, 15)];

**return** $color;

}

**function** Barnsley($x, $y, $C, $centru\_x, $centru\_y){ // functia calculeaza si salveaza coordonatele unui punct ce apartine fractalului

**global** $M; // vom salva in matricea M toate punctele care fac parte din fractalul Barnsley

$numar = **rand**(1, 100);

$M[**intval**($GLOBALS['px']\*$GLOBALS['zoom']+500-$centru\_x)][**intval**($GLOBALS['py']\*(-1)\*$GLOBALS['zoom']+650-$centru\_y)] = 1; // aplicand formulele descrise in aceasta linie

**if**($numar <= $GLOBALS['p1']){ // urmatoarele linii salveaza in px si py coordonatele noului punct din fractal folosind algoritmul de generare al unui fractal Barnsley

$GLOBALS['px'] = $C[1][1]\*$x + $C[1][2]\*$y + $C[1][5];

$GLOBALS['py'] = $C[1][3]\*$x + $C[1][4]\*$y + $C[1][6];

}

**else** **if**($numar <= $GLOBALS['p1']+$GLOBALS['p2']){

$GLOBALS['px'] = $C[2][1]\*$x + $C[2][2]\*$y + $C[2][5];

$GLOBALS['py'] = $C[2][3]\*$x + $C[2][4]\*$y + $C[2][6];

}

**else** **if**($numar <= $GLOBALS['p1']+$GLOBALS['p2']+$GLOBALS['p3']){

$GLOBALS['px'] = $C[3][1]\*$x + $C[3][2]\*$y + $C[3][5];

$GLOBALS['py'] = $C[3][3]\*$x + $C[3][4]\*$y + $C[3][6];

}

**else** **if**($numar <= $GLOBALS['p1']+$GLOBALS['p2']+$GLOBALS['p3']+$GLOBALS['p4']){

$GLOBALS['px'] = $C[4][1]\*$x + $C[4][2]\*$y + $C[4][5];

$GLOBALS['py'] = $C[4][3]\*$x + $C[4][4]\*$y + $C[4][6];

}

}

**if**(**isset**($\_POST['Galerie'])){ // daca s-a accesat GALERIA

**echo** '<div class="centru"> <img src="Imagini/Galerie1.png" width=900 alt=""> </div> <br>'; // afisam imaginea GALERIE la centru

**echo** '<div class="centru"> <form action="Index.html"> <input type="submit" value="Inapoi la prima pagina"></form></div><br><br>'; // si un formular care ne trimite inapoi la pagina principala

**for**($i = 0; $i<=3; $i++){ // vom afisa toate pozele salvate: pentru fiecare din cele 4 tipuri de fractali

**echo** "<p class='tip'> -- ",$directoare[$i], " --</p>"; // afisam intr-un paragraf " -- TIP FRACTAL -- "

**echo** "<div class='scrollmenu'>"; // deschidem o bara orizontala de scroll

$director = './Exemple\_Frunze/'.$directoare[$i].'/'; // creem adresa fisierului in care se afla imaginile fractalilor ce prezinta tipul curent

**if**($deschide\_director = **opendir**($director)){ // daca s-a deschis fisierul

**while**(($fisier = **readdir**($deschide\_director)) !== **FALSE**){ // cat timp exista fisiere de citit

**if**($fisier != "." && $fisier != ".."){ // daca fisierul este diferit de 0 -- adica daca este o imagine

**echo** "<img src='**$director**/**$fisier**' class='img1' alt=''>"; // adaugam la scrollbar imaginea respectiva

}

}

}

**echo** "</div>"; // inchidem bara orizontala de scroll

**echo** "<br><br>";

}

}

**else**{ // altfel, daca nu accesam GALERIA, vom genera un fractal

**echo** '<div class="centru"> <img src="Imagini/Imagine\_Generare.png" width=900 alt=""> </div> <br>'; // afisam imaginea GENERARE

**echo** '<div class="centru"> <button class="button" id="Gata" onclick="arata\_imagine()"> Vezi direct imaginea </button> <form action="Index.html"> <input type="submit" value="Inapoi la prima pagina"></form></div><br><br>'; // cream butonul care va afisa direct imaginea generata in functie de optiunile alese

**echo** '<div class="centru"><canvas id="plansa"> </canvas></div><br><br>'; // cream o plansa la mijlocul paginii unde vom arata cum este creat fractalul

**if**(**isset**($\_POST['Random\_colors'])){ // daca s-a ales sa se genereze un fractal de culori aleatorii

$tip = $\_POST['tip']; // salvam tipul ales de fractal

$culoare\_frunza = random\_hex\_color(); // generam 3 culori aleatorii pentru frunza, fundalul si umbra fractalului

$culoare\_fundal = random\_hex\_color();

$culoare\_umbra = random\_hex\_color();

**while**($culoare\_frunza == $culoare\_fundal || $culoare\_fundal == $culoare\_umbra || $culoare\_umbra == $culoare\_frunza){ // atat timp cat exista 2 culori identice

$culoare\_frunza = random\_hex\_color(); // generam alte 3 culori aleatorii

$culoare\_fundal = random\_hex\_color();

$culoare\_umbra = random\_hex\_color();

}

$calitate = $\_POST['calitate']; // salvam calitatea imaginii in functie de care vom afla numarul de puncte din care va fi format fractalul

**if**($calitate == "Simplu"){ // calitatea SIMPLU corespunde numarului calitate[0] = 400000 puncte, MEDIU la calitate[1] = 700000 si COMPLEX la 1000000

$numar = $numar\_puncte[0];

}

**else** **if**($calitate == "Mediu"){

$numar = $numar\_puncte[1];

}

**else**{

$numar = $numar\_puncte[2];

}

}

**else** **if**(**isset**($\_POST['Random\_fractal'])){ // daca s-a ales sa se genereze un fractal complet aleator

$i = **rand**(0, 255)%4; // alegem un numar aleator de la 0 la 3

$tip = $directoare[$i]; // numarul respectiv va fi cel ce va determina tipul fractalului

$culoare\_frunza = random\_hex\_color(); // generam 3 culori aleatorii pentru frunza, umbra si fundalul fractalului

$culoare\_fundal = random\_hex\_color();

$culoare\_umbra = random\_hex\_color();

**while**($culoare\_frunza == $culoare\_fundal || $culoare\_fundal == $culoare\_umbra || $culoare\_umbra == $culoare\_frunza){ // cat timp exista 2 culori identice

$culoare\_frunza = random\_hex\_color(); // vom genera alte 3 culori aleatorii

$culoare\_fundal = random\_hex\_color();

$culoare\_umbra = random\_hex\_color();

}

$i = **rand**(0, 255)%3; // alegem un numar aleator de la 0 la 2

$numar = $numar\_puncte[$i]; // numarul respectiv va fi cel ce va determina numarul de puncte al fractalului

}

**else** **if**(**isset**($\_POST['Genereaza'])){ // daca s-a ales sa se genereze un fractal in functie de parametrii alesi

$culoare\_frunza = $\_POST['culoare\_frunza']; // salvam in variabile culorile, tipul si numarul de puncte al fractalului

$culoare\_fundal = $\_POST['culoare\_fundal'];

$culoare\_umbra = $\_POST['culoare\_umbra'];

$tip = $\_POST['tip'];

$calitate = $\_POST['calitate'];

**if**($calitate == "Simplu"){

$numar = $numar\_puncte[0];

}

**else** **if**($calitate == "Mediu"){

$numar = $numar\_puncte[1];

}

**else**{

$numar = $numar\_puncte[2];

}

}

$nume\_fisier\_coeficienti = "./Valori\_Coeficienti/".$tip.".txt"; // formam calea fisierului pentru a citi coeficientii

$citeste = **fopen**($nume\_fisier\_coeficienti, "r"); // deschidem fisierul pentru a citi numere reale

$total = 0;

**if**($citeste){ // daca s-a deschis fisierul

**while**(($valoare = **fgets**($citeste, 1000)) != **false**){ // cat timp exista linii de citit

$p = **strtok**($valoare, " "); // separam linia citita in parti delimitate de caracterul " "

**while**($p != **NULL**){ // cat timp exista " "

$coeficienti[$total] = **floatval**($p); // salvam coeficientul in vectorul respectiv

$total ++; // marim numarul de coeficienti cu 1

$p = **strtok**(" "); // cautam din nou caracterul " "

}

}

**fclose**($citeste); // citirea s-a terminat

}

$i = -1;

$n = $coeficienti[++$i]; // folosind vectorul "coeficienti" copiem valorile pentru a crea variabile noi

$m = $coeficienti[++$i];

$centru\_x = $coeficienti[++$i];

$centru\_y = $coeficienti[++$i];

$zoom = $coeficienti[++$i];

$p1 = $coeficienti[++$i];

$p2 = $coeficienti[++$i];

$p3 = $coeficienti[++$i];

$p4 = $coeficienti[++$i];

$indice = ++$i;

**for**($i = 1; $i<=4; $i++){ // cream matricea de coeficienti descrisa in modul de generare al fractalului Barnsley

**for**($j = 1; $j<=6; $j++){

$C[$i][$j] = $coeficienti[$indice];

$indice++;

}

}

**for**($i = 0; $i < $n; $i++){ // curatam matricea in care vom salva punctele ce fac parte din fractal

**for**($j = 0; $j < $m; $j++){

$M[$i][$j] = 0;

}

}

**for**($i = 1; $i<=$numar; $i++){ // generam si salvam in matricea M punctele ce formeaza fractalul

Barnsley($px, $py, $C, $centru\_x, $centru\_y);

}

**for**($i = 0; $i<$n; $i++){ // parcurgem matricea M pentru a determina ce puncte formeaza umbra fractalului

**for**($j = 0; $j<$m; $j++){

**for**($k = 0; $k<4 && $M[$i][$j] == 1; $k++){

$l9 = $i + $dl[$k];

$c9 = $j + $dc[$k];

**if**($M[$l9][$c9] == 0){ // daca un punct care nu face parte din fractal este vecinul unui punct ce e parte din fractal

$M[$l9][$c9] = 2; // atunci punctul respectiv face parte din umbra fractalului

}

}

}

}

$imagine = **imagecreatetruecolor**($n, $m); // creem o variabila in care vom desena fractalul

**for**($i = 0; $i<$n; $i++){ // parcurgem matricea M

**for**($j = 0; $j<$m; $j++){

**if**($M[$i][$j] == 1){ // daca M[i][j] este 1

$color = $culoare\_frunza; // punctul respectiv formeaza frunza fractalului

}

**else** **if**($M[$i][$j] == 2){ // altfel, daca e 2

$color = $culoare\_umbra; // punctul face parte din umbra fractalului

}

**else**{ // altfel, punctul face parte din fundal

$color = $culoare\_fundal;

}

**imagesetpixel**($imagine, $i, $j, **hexdec**($color)); // desenam la pozitia (i,j) in imagine punctul de culoarea respectiva

}

}

$fisiere = 0; // vom forma numele imaginii cu fractalul pe care il vom genera

$director = './Exemple\_Frunze/'.$tip.'/'; // deschidem fisierul cu tipul fractalului ales

**if**($deschide\_director = **opendir**($director)){ // daca s-a deschis fisierul

**while**(($fisier = **readdir**($deschide\_director)) !== **FALSE**){ // cat timp exista fisiere de citit

**if**($fisier != "." && $fisier != ".."){ // daca numele fisierului nu e nul

$fisiere++; // actualizam numarul de fisiere gasite

}

}

}

$fisiere++; // incrementam numarul de fisiere gasite - numarul lor va fi mai mare cu 1 deoarece imaginea pe care o vom crea se va afla aici

$nume\_fisier = './Exemple\_Frunze/'.$tip.'/'.$tip.$fisiere.".png"; // cream adresa si numele imaginii cu fractalul generat din tipul sau + numarul pozei

**imagepng**($imagine, $nume\_fisier); // generam poza la adresa creata mai sus

**echo** "<p class='tip' hidden> <img id='imagine\_creata' src='**$nume\_fisier**' class='img1' alt=''> </p><br><br>"; // afisam intr-un paragraf ascuns imaginea cu fractalul generat

**if**(!**isset**($\_SESSION["nume\_fisier"])){ // salvam in sesiune variabilele pe care le vom folosi in sursa JAVASCRIPT

$\_SESSION["nume\_fisier"] = $nume\_fisier;

}

**if**(!**isset**($\_SESSION["tip"])){

$\_SESSION["tip"] = $tip;

}

**if**(!**isset**($\_SESSION["fisiere"])){

$\_SESSION["fisiere"] = $fisiere;

}

**if**(!**isset**($\_SESSION["coeficienti"])){

$\_SESSION["coeficienti"] = $coeficienti;

}

**if**(!**isset**($\_SESSION["culoare\_frunza"])){

$\_SESSION["culoare\_frunza"] = $culoare\_frunza;

}

**if**(!**isset**($\_SESSION["culoare\_fundal"])){

$\_SESSION["culoare\_fundal"] = $culoare\_fundal;

}

**if**(!**isset**($\_SESSION["culoare\_umbra"])){

$\_SESSION["culoare\_umbra"] = $culoare\_umbra;

}

**if**(!**isset**($\_SESSION["numar"])){

$\_SESSION["numar"] = $numar;

}

**if**(!**isset**($\_SESSION["M"])){

$\_SESSION["M"] = $M;

}

}

?><!-- inchidem sursa PHP -->

<script> <!-- deschidem o sursa JAVASCRIPT -->

***var*** coeficienti **=** <?php **echo** **json\_encode**($coeficienti); ?>**;** // cream variabile pentru valorile salvate in sesiune si le salvam ca in sursa precedenta

***var*** culoare\_umbra **=** <?php **echo** **json\_encode**($culoare\_umbra); ?>**;**

***var*** culoare\_fundal **=** <?php **echo** **json\_encode**($culoare\_fundal); ?>**;**

***var*** culoare\_frunza **=** <?php **echo** **json\_encode**($culoare\_frunza); ?>**;**

***var*** numar\_puncte **=** <?php **echo** **json\_encode**($numar); ?>**;**

***var*** M **=** <?php **echo** **json\_encode**($M); ?>**;**

***var*** nume\_fisier **=** <?php **echo** **json\_encode**($nume\_fisier); ?>**;**

***var*** dl **=** **[-**1**,** 0**,** 1**,** 0**];**

***var*** dc **=** **[**0**,** 1**,** 0**,** **-**1**];**

***var*** C **=** **[];**

***var*** i **=** **-**1**;**

***var*** j**;**

***var*** px **=** 0**;**

***var*** py **=** 0**;**

***var*** n **=** coeficienti**[++**i**];**

***var*** m **=** coeficienti**[++**i**];**

***var*** centru\_x **=** coeficienti**[++**i**];**

***var*** centru\_y **=** coeficienti**[++**i**];**

***var*** zoom **=** coeficienti**[++**i**];**

***var*** p1 **=** coeficienti**[++**i**];**

***var*** p2 **=** coeficienti**[++**i**];**

***var*** p3 **=** coeficienti**[++**i**];**

***var*** p4 **=** coeficienti**[++**i**];**

***var*** indice **=** **++**i**;**

***for*(**i **=** 1**;** i**<=**4**;** i**++){**

C**[**i**]** **=** ***new*** Array**();**

***for*(**j **=** 1**;** j**<=**6**;** j**++){**

C**[**i**][**j**]** **=** coeficienti**[**indice**];**

indice**++;**

**}**

**}**

C**[**0**]** **=** ***new*** Array**();**

***for*(**i **=** 0**;** i**<=**4**;** i**++){**

C**[**i**][**0**]** **=** 0**;**

C**[**0**][**i**]** **=** 0**;**

**}**

***function*** arata\_imagine**(){** // functia va afisa direct imaginea creata a fractalului de generat

context.fillStyle **=** culoare\_fundal**;** // setam culoarea curenta cu cea a fundalului fractalului

context.fillRect**(**0**,**0**,** n**,** m**);** // coloram toata plansa folosind culoarea respectiva

***var*** imagine **=** document.getElementById**(**"imagine\_creata"**);** // accesam imaginea creata in sursa PHP si salvata in paragraful ascuns

context.drawImage**(**imagine**,** 0**,** 0**);** // desenam imaginea

***var*** buton **=** document.getElementById**(**"Gata"**);** // ascundem butonul "Vezi direct imaginea"

buton.style.display **=** "none"**;**

**}**

<?php

**if**(!**isset**($\_POST["Galerie"])){ // daca nu s-a accesat galeria

**echo** '</script>'; // inchidem sursa

**echo** '<script src="Generare\_Animata.js"> </script>'; // rulam sursa ce va arata cum se genereaza fractalul Barnsley

}

?>

</body><!-- inchidem corpul paginii -->

</html>

Fișierul *Generare\_Animata.js*:

**var** canvas **=** **document.**getElementById**(**"plansa"**);** // accesam plansa creata pe care vom desena fractalul Barnsley

**var** context **=** canvas**.**getContext**(**"2d"**);**

**var** numar **=** 0**;**

canvas**.**width **=** n**;**

canvas**.**height **=** m**;**

context**.**fillStyle **=** culoare\_fundal**;** // setam culoarea de desenare la cea a fundalului ales pentru fractal

context**.**fillRect**(**0**,**0**,** n**,** m**);** // desenam fundalul fractalului

**function** Barnsley**(**x1**,** y1**){** // cream fractalul ca in sursa "Creeaza\_Fractal.php"

numar **=** Math**.**floor**(**Math**.**random**()\***101**);**

**var** yy **=** **parseInt(**py**\*(-**1**)\***zoom **+** 650 **-** centru\_y**);** // cream coordonatele la care vom desena pixelul din imagine

**var** xx **=** **parseInt(**px**\***zoom **+** 500 **-** centru\_x**);**

context**.**fillStyle **=** culoare\_frunza**;** // setam culoarea de desenare la cea a frunzei

context**.**fillRect**(**xx**,** yy**,** 1**,** 1**);** // desenam pixelul la coordonatele respective

**if(**numar **<=** p1**){** // generam urmatorul punct din fractal

px **=** C**[**1**][**1**]\***x1 **+** C**[**1**][**2**]\***y1 **+** C**[**1**][**5**];**

py **=** C**[**1**][**3**]\***x1 **+** C**[**1**][**4**]\***y1 **+** C**[**1**][**6**];**

**}**

**else** **if(**numar **<=** p1 **+** p2**){**

px **=** C**[**2**][**1**]\***x1 **+** C**[**2**][**2**]\***y1 **+** C**[**2**][**5**];**

py **=** C**[**2**][**3**]\***x1 **+** C**[**2**][**4**]\***y1 **+** C**[**2**][**6**];**

**}**

**else** **if(**numar **<=** p1 **+** p2 **+** p3**){**

px **=** C**[**3**][**1**]\***x1 **+** C**[**3**][**2**]\***y1 **+** C**[**3**][**5**];**

py **=** C**[**3**][**3**]\***x1 **+** C**[**3**][**4**]\***y1 **+** C**[**3**][**6**];**

**}**

**else** **if(**numar **<=** p1 **+** p2 **+** p3 **+** p4**){**

px **=** C**[**4**][**1**]\***x1 **+** C**[**4**][**2**]\***y1 **+** C**[**4**][**5**];**

py **=** C**[**4**][**3**]\***x1 **+** C**[**4**][**4**]\***y1 **+** C**[**4**][**6**];**

**}**

**}**

**function** deseneaza**(){**

Barnsley**(**px**,** py**);**

**}**

**function** umbra**(){** // cream umbra fractalului folosindu-ne de matricea M

context**.**fillStyle **=** culoare\_umbra**;**

**for(**i **=** 0**;** i**<**n**;** i**++){**

**for(**j **=** 0**;** j**<**m**;** j**++){**

**if(**M**[**i**][**j**]** **==** 2**){**

context**.**fillRect**(**i**,** j**,** 1**,** 1**);**

**}**

**}**

**}**

**}**

**for(**i **=** 1**;** i**<=**numar\_puncte**;** i**++){** // la fiecare milisecunda vom desena un punct

**setInterval(**deseneaza**,** 1**);**

**}**

**setInterval(**umbra**,** numar\_puncte**/**1000**);** // dupa ce va fi generat fractalul, vom desena si umbra acestuia

Fișierul *Design1.css*:

body **{**

**background-image: url(Imagini/BackgroundF.jpg);**

**background-size: cover;**

**background-repeat: repeat-x;**

**background-attachment: fixed;**

**animation: make\_a\_change 60s linear infinite;**

**}**

**@**keyframes make\_a\_change **{**

from**{**

**background-position: 0 0;**

**}**

to**{**

**background-position: 100% 0;**

**}**

**}**

pre**,** p**,** center**{**

**background-color: rgba(255, 255, 255, 0.6);**

**color: rgba(0, 0, 0, 1);**

**font-weight: bold;**

**width: 100%;**

**text-align: center;**

**}**

div**.**centru**{**

**text-align: center;**

**background-color: rgba(255, 255, 255, 0.6);**

**color: rgba(0, 0, 0, 1);**

**font-weight: bold;**

**width: 100%;**

**}**

p**.**central**{**

**text-align: center;**

**background-color: rgba(255, 255, 255, 0.6);**

**color: rgba(0, 0, 0, 1);**

**font-weight: bold;**

**width: 100%;**

**}**

**.**button **{**

**background-color: rgba(0, 59, 3, 1);**

**color: white;**

**cursor: pointer;**

**padding: 16px 32px;**

**font-size: 15px;**

**text-decoration: bold;**

**text-align: center;**

**border-radius: 20px;**

**}**

**.**button**:hover{**

**background-color: rgba(0, 0, 0, 1);**

**color: rgba(255, 255, 255, 1);**

**text-decoration: bold;**

**text-align: center;**

**}**

input**[**type=submit**]{**

**background-color: rgba(0, 59, 3, 1);**

**color: white;**

**cursor: pointer;**

**padding: 16px 32px;**

**font-size: 15px;**

**text-decoration: bold;**

**text-align: center;**

**border-radius: 20px;**

**}**

input**[**type=submit**]:hover{**

**background-color: rgba(0, 0, 0, 1);**

**color: rgba(255, 255, 255, 1);**

**text-decoration: bold;**

**text-align: center;**

**}**

select**{**

**background-color: rgba(0, 59, 3, 1);**

**color: white;**

**text-align: center;**

**border-radius: 20px;**

**width: 20%;**

**margin: 6px;**

**}**

select**:hover{**

**background-color: rgba(0, 0, 0, 1);**

**color: white;**

**}**

option**{**

**background-color: rgba(0, 0, 0, 1);**

**color: white;**

**}**

img **{**

**width: 100%;**

**height: auto;**

**}**

Fișierul Design2.css:

canvas **{**

**border: dashed;**

**border-width: 5px;**

**margin-right: 10px;**

**margin-bottom: 5px;**

**margin-top: 5px;**

**border-color: rgba(255, 255, 255, 1);**

**}**

div**.**scrollmenu **{**

**text-align: center;**

**background-color: rgba(0, 0, 0, 0.8);**

**overflow-x: scroll;**

**white-space: nowrap;**

**}**

div**.**scrollmenu img **{**

**border: dashed;**

**border-width: 5px;**

**margin-right: 10px;**

**margin-bottom: 5px;**

**margin-top: 5px;**

**border-color: rgba(255, 255, 255, 1);**

**display: inline-block;**

**text-align: center;**

**}**

div**.**scrollmenu img**:hover** **{**

**border-color: rgba(3, 200, 0, 1);**

**}**

body **{**

**background-image:url(Imagini/BackgroundF.jpg);**

**background-size: cover;**

**background-attachment: fixed;**

**animation-name: make\_a\_change1;**

**animation-duration: 120s;**

**animation-iteration-count: infinite;**

**width: auto;**

**}**

**@**keyframes make\_a\_change1 **{**

0% **{**

**background-position: 0% 0%;**

**}**

25% **{**

**background-position: -25% 0%;**

**}**

50% **{**

**background-position: -50% 0%;**

**}**

75% **{**

**background-position: -75% 0%;**

**}**

100% **{**

**background-position: -100% 0%;**

**}**

**}**

href **{**

**text-decoration: none;**

**}**

a**:link** **{**

**background-color: rgba(0, 59, 3, 1);**

**color: white;**

**cursor: pointer;**

**padding: 16px 32px;**

**font-size: 15px;**

**text-decoration: bold;**

**text-align: center;**

**border-radius: 20px;**

**}**

a**:hover** **{**

**background-color: rgba(0, 0, 0, 1);**

**color: rgba(255, 255, 255, 1);**

**text-decoration: bold;**

**text-align: center;**

**}**

p**,** center **{**

**background-color: rgba(255, 255, 255, 0.8);**

**font-size: 20px;**

**font-weight: bold;**

**color: rgba(0, 59, 3, 1);**

**width: auto;**

**}**

p**.**tip **{**

**text-align: center;**

**background-color: rgba(0, 0, 0, 0.8);**

**font-size: 20px;**

**font-weight: bold;**

**color: rgba(255, 255, 255, 1);**

**}**

p**.**tip1 **{**

**text-align: center;**

**}**

img**.**img1**{**

**border: dashed;**

**border-width: 5px;**

**margin-right: 10px;**

**margin-bottom: 5px;**

**margin-top: 5px;**

**border-color: rgba(255, 255, 255, 1);**

**width: auto;**

**}**

img**.**img1**:hover** **{**

**border-color: rgba(3, 200, 0, 1);**

**}**

Ultimele 2 fișiere sunt folosite pentru a descrie elementele paginii “Barnsley Generator”, ele având rolul de a caracteriza cum anume să arate anumite taguri din pagină.