

```

1.  /* Objeto de Datos */
2.  var datos = {
3.      fun: [],
4.      xi: [],
5.      fi: [],
6.      h: 0.001,
7.      i: 0,
8.      tam: {n:3,m:3}
9.  };
10. function initialEvents(){
11.     $('#btnCalcular').click(function(){
12.         init();
13.         $('#tablasDer').html("");
14.         calcSteffensen();
15.     });
16.     $('#btnReset').click(function(){
17.         $('#tablasDer').html("");
18.         resetDatos();
19.     });
20.     $('#btnDefault').click(function(){
21.         $('#tablasDer').html("");
22.         resetDatos();
23.         defaultDatos();
24.     });
25. }
26.
27. /* Funciones */
28. function defaultDatos(){
29.     $('#initialX1').val('5');
30.     $('#initialX2').val('2');
31.     $('#initialF1').val('function(x1,x2){ return (x1*x1)-2*(x1*x2)+x2-8 }');
32.     $('#initialF2').val('function(x1,x2){ return x1+2*(x2*x2)-10 }');
33. }
34. function resetDatos(){
35.     /* reseteando matriz Xo */
36.     var initialVar = document.forms.initialVar.children;
37.     [].forEach.call(initialVar,function(i,index){
38.         i.children[1].value = "";
39.     });
40.     /* reseteando matriz de funciones */
41.     var initialFun = document.forms.initialFun.children;
42.     [].forEach.call(initialFun,function(i,index){
43.         i.children[1].value = "";
44.     });
45.     datos = {
46.         fun: [],
47.         xi: [],
48.         fi: [],
49.         h: 0.001,
50.         i: 0,
51.         tam: {n:3,m:3},
52.         k: [],
53.     };
54. }
55. function init(){
56.     var cont = 0;
57.     /* Obteniendo matriz Xo */
58.     var initialVar = document.forms.initialVar.children;
59.     [].forEach.call(initialVar,function(i,index){
60.         if(i.children[1].value == ""){
61.             cont++;
62.         }else{
63.             datos.xi[index] = parseFloat(i.children[1].value);
64.         }
65.     });
66.

```

```

67.     /* Obteniendo matriz de funciones */
68.     var initialFun = document.forms.initialFun.children;
69.     [].forEach.call(initialFun,function(i,index){
70.         if(i.children[1].value == ""){
71.             cont++;
72.         }else{
73.             datos.fun[index] = eval('(' + i.children[1].value + ')');
74.         }
75.     });
76.     /* Resolver las funciones con los valores iniciales Matriz Fo*/
77.     calcFun();
78.
79.     if(cont>0){
80.         $('#msg').text('Le falta llenar datos');
81.     }
82.     else{
83.         $('#msg').text('');
84.     }
85. }
86.
87. function calcFun(){
88.     datos.xi.forEach(function(i,index){
89.         datos.fi[index] = datos.fun[index].apply(this,datos.xi);
90.     });
91.     mostrarHead();
92.     mostrarMatrizX();
93.     mostrarFunciones();
94. }
95. function calcMatrizK(){
96.     var x1 = datos.xi[0];
97.     var x2 = datos.xi[1];
98.     var matrizK = new Array();
99.     for(var i=0;i<datos.tam.m-1;i++){
100.         matrizK[i] = new Array();
101.         for(var j=0;j<datos.tam.n-1;j++){
102.             if(!j%2){
103.                 matrizK[i].push( (datos.fun[i](x1+datos.fi[i], x2) - datos.fi[i])/ d
atos.fi[i] );
104.             }
105.             else{
106.                 matrizK[i].push( (datos.fun[i](x1, x2 + datos.fi[i]) - datos.fi[i])/
datos.fi[i] );
107.             }
108.         }
109.     }
110.     mostrarMatrizK(matrizK);
111.     return matrizK;
112. }
113. function calcInversaMatrizK(){
114.     var matrizK = new Matrix(calcMatrizK());
115.     var inversaMatrizK = matrizK.clone().inverse();
116.     return inversaMatrizK;
117. }
118.
119. function calcSteffensen(){
120.     var k;
121.     var f0;
122.     var x0 = new Matrix(new Array(datos.xi));
123.     var m1 = [];
124.     var xn = [];
125.     var cont = 0;
126.
127.     while(cont<10){
128.         calcFun();
129.         k = calcInversaMatrizK();
130.         f0 = new Matrix(new Array(datos.fi));

```

```

131.
132.         m1 = f0.clone().multiply(k.clone());
133.         xn = m1.clone().subtract(x0.clone());
134.         if(xn.equals(x0)){
135.             break;
136.         }
137.         else{
138.             x0 = xn.clone();
139.             datos.xi[0] = x0[0][0];
140.             datos.xi[1] = x0[0][1];
141.         }
142.         cont++;
143.         datos.i = cont;
144.     }
145.     var html = "<h3>Resultado x"+datos.i+"</h3><table class='table table-
striped table-hover table-bordered'><tbody>";
146.     for(var i=0;i<xn.cols;i++){
147.         html+= " <tr><td>"+xn[0][i].toPrecision(4)+"</td></tr>";
148.     }
149.     html+="</tbody></table>";
150.     $('#tablasDer').append(html);
151. }
152.
153. /* mostrar datos */
154. function mostrarHead(){
155.     var html = "<br/><h1 class='text-
primary'>Iteracion "+(datos.i+1)+"</h1>";
156.     $('#tablasDer').append(html);
157. }
158. function mostrarMatrizX(){
159.     var m = new Matrix(new Array(datos.xi));
160.     var html = "<table class='table table-striped table-hover table-
bordered'><thead class='thead-
dark'><tr><th colspan='"+m.cols+"'>matriz X"+datos.i+"</th></tr></thead><tbody>";
161.     for(var i=0; i<m.cols; i++){
162.         html+= " <tr><td>"+m[0][i]+"</td></tr>";
163.     }
164.     html+="</tbody></table>";
165.     $('#tablasDer').append(html);
166. }
167. function mostrarFunciones(){
168.     var m = new Matrix(new Array(datos.fi));
169.     var html = "<h6>matriz f"+datos.i+"</h6><table class='table table-
striped table-hover table-bordered'><tbody>";
170.     for(var i=0; i<m.cols; i++){
171.         html+= " <tr><td>"+m[0][i]+"</td></tr>";
172.     }
173.     html+="</tbody></table>";
174.     $('#tablasDer').append(html);
175. }
176. function mostrarMatrizK(matrizK){
177.     var m = new Matrix(matrizK);
178.     var html = "<h6>Calculo de matriz K"+datos.i+"</h6><table class='table tab
le-striped table-hover table-bordered'><tbody>";
179.
180.     for(var i=0; i<m.rows; i++){
181.         html+= '<tr>';
182.         for(var j=0; j<m.cols; j++){
183.             html+= " <td>"+m[i][j]+"</td>";
184.         }
185.         html+= '</tr>';
186.     }
187.     html+="</tbody></table>";
188.     $('#tablasDer').append(html);
189. }
190. /* Inicializar Eventos */

```

```
191.         initialEvents();
```