

Contenido: Introducción a la Informática

LAURA QUINTERO MONTOYA
30 DE SEPTIEMBRE DE 2020

1 CONTENIDO

1	CONTENIDO.....	1
2	PRESENTACIÓN.....	2
3	Función Numérica	3
3.1	Sistema Binario.....	4
4	Bibliografía	14

2 PRESENTACIÓN

La presente monografía describe como la materia introducción a la informática busca y estudia varios campos del conocimiento ligados a la teoría de la Informática y los Sistemas en general. Se han identificado varias áreas que representan el sustento teórico y práctico de la carrera, según se ha mencionado en el resumen del documento.

El objetivo del presente documento es describir cada uno de los temas mencionados, lo cual le permitirá obtener conocimientos a la hora de desarrollar programas o trabajos que requiera la carrera en general.

3 FUNCIÓN NUMÉRICA

Una función numérica no necesariamente tiene un conjunto de números como dominio, como es el caso de la probabilidad; sin embargo, su imagen es siempre un conjunto de números. Por Ejemplos:

La relación que asocia un número entero con el doble de su valor es una función numérica. Se puede definir de la siguiente manera: $f: \mathbb{N} \rightarrow \mathbb{N} \mid f(x) = 2x$

En el caso de que si hablamos de programación: En los lenguajes de programación, las funciones numéricas calculan valores de "funciones" tal como se definen en la ciencia de las matemáticas. Mientras que las funciones en los lenguajes de programación son subrutinas que devuelven valores dados ciertos parámetros, las funciones en matemáticas se definen como relaciones entre elementos en un conjunto de definición D_f , y elementos en un conjunto de valores D_v , de manera que:

$f: x \rightarrow y$, donde x pertenece a D_f y y a D_v ,

Las funciones matemáticas de los lenguajes de programación se adhieren a la regla matemática del mismo valor de retorno cada vez de las funciones matemáticas. Los más utilizados producen un valor de retorno de punto flotante a partir de uno o, a veces, dos argumentos de punto flotante. Las funciones matemáticas más comunes se refieren a raíces cuadradas, logaritmos, exponenciales y trigonometría.

3.1 SISTEMA BINARIO

Sistema numérico binario, en matemáticas, sistema numérico posicional que emplea 2 como base y, por lo tanto, requiere solo dos símbolos diferentes para sus dígitos, 0 y 1, en lugar de los 10 símbolos diferentes habituales que se necesitan en el sistema decimal. Los números del 0 al 10 están, por tanto, en 0, 1, 10, 11, 100, 101, 110, 111, 1000, 1001 y 1010 binarios. La importancia del sistema binario para la teoría de la información y la tecnología informática se deriva principalmente del compacto y una manera confiable en la que los 0 y los 1 se pueden representar en dispositivos electromecánicos con dos estados, como "encendido-apagado", "abierto-cerrado" o "pasa-no pasa". (Ver numerales y sistemas de numeración: El sistema binario).

Debido a que solo hay dos valores booleanos válidos para representar un "1" lógico o un "0" lógico, hace que el sistema de uso de números binarios sea ideal para su uso en circuitos y sistemas digitales o electrónicos.

A continuación se demostrara como se utiliza el sistema binario:

Primero se crea un nuevo código en el html, estructurando nuestro Head y body ; y se crea y llama la función **sistema_binario()**:



```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="utf-8">
5      <meta name="viewport" content="width=device-width">
6      <title>repl.it</title>
7      <link href="style.css" rel="stylesheet" type="text/css" />
8    </head>
9    <body onload="sistema_binario()">
10     <script src="script.js"></script>
11  </body>
12 </html>
```

Grafica 1. Código 1

Dentro de esa función se agrega la variable **document.write**, que esta tiene la función de escribir en la página, agregando las comillas ("") y "br" que este nos ayuda a dar espacios entre los párrafos. En este caso se define el título, el texto y los ejemplos. En el tema es el sistema binario, su definición y como se representa los números normales en este código

```
1 function sistema_binario()  
2   document.write("EL SISTEMA BINARIO <br/><br/>");  
3  
4   document.write("Está constituido por un modelo de  
operaciones matemáticas sobre dos representaciones  
numéricas únicas: El UNO y el CERO. El número UNO se  
representa con el cardinal 1, mientras que el número CERO  
se representa con el cardinal 0. Todos los números  
decimales se representan a partir de combinaciones únicas  
de números binarios <br/><br/>");  
5  
6   document.write("Los primeros cuatro número binarios se  
representan así: <br/><br/>");  
7  
8   document.write("0 => 00 <br/>");  
9   document.write("1 => 01 <br/>");  
10  document.write("2 => 10 <br/>");  
11  document.write("3 => 11 <br/>");  
12
```

Grafica 2. Código 1

Y así se debería visualizar en la página:

EL SISTEMA BINARIO

Está constituido por un modelo de operaciones matemáticas sobre dos representaciones numéricas únicas: El UNO y el CERO. El número UNO se representa con el cardinal 1, mientras que el número CERO se representa con el cardinal 0. Todos los números decimales se representan a partir de combinaciones únicas de números binarios

Los primeros cuatro número binarios se representan así:

0 => 00
1 => 01
2 => 10
3 => 11

Grafica 3. Código completo

En el segundo programa crea una nueva función “numero_binario_0_7()”

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width">
6     <title>repl.it</title>
7     <link href="style.css" rel="stylesheet" type="text/css" />
8   </head>
9   <body onload="numero_binario_0_7()">
10    <script src="script.js"></script>
11  </body>
12 </html>
```

Grafica 4. Código nuevo.

Y en su función se coloca un nuevo texto, en este caso se muestra como es el código binario del 0 al 7, todo eso con la variable “document.write”:

```
1 function numero_binario_0_7(){
2   document.write("NÚMEROS BINARIOS DEL 0 AL 7 <br/><br/>");
3
4   document.write("0 => 000 <br/>");
5   document.write("1 => 001 <br/>");
6   document.write("2 => 010 <br/>");
7   document.write("3 => 011 <br/>");
8   document.write("4 => 100 <br/>");
9   document.write("5 => 001 <br/>");
10  document.write("6 => 010 <br/>");
11  document.write("7 => 111 <br/>");
12 }
```

Grafica 5. Código del texto “números binarios del 0 al 7”.

Como se muestra al final del código, los números binarios tiene su patrón, el UNO se mueve dependiendo de del número que se desea obtener:

NÚMEROS BINARIOS DEL 0 AL 7

0 => 000
1 => 001
2 => 010
3 => 011
4 => 100
5 => 001
6 => 010
7 => 111

Grafica 6. Código completo del programa 2

En el tercer programa, se crea la misma función de “sistema_binario()”:

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="utf-8">
5      <meta name="viewport" content="width=device-width">
6      <title>repl.it</title>
7      <link href="style.css" rel="stylesheet" type="text/css"
8      />
9    </head>
10   <body onload="numeros_binarios()">
11     <script src="script.js"></script>
12   </body>
13 </html>
```

Grafica 6. Código función del programa 3

Ya en su body se cambia su título, párrafo y ejemplos, explicando que los decimales al binario y que la combinación de cada número decimal, garantiza el cambio suyo respecto a los demás:


```
1 function numeros_binarios() {  
2     document.write("DECIMAL A BINARIO (0 - 15) <br/><br/>");  
3     document.write("La combinación de números binarios debe  
4         ser diferente para cada número decimal. Esto garantiza  
5         que cada número es diferente de los demás. <br/><br/>");  
6     document.write("0 ==> 0000 <br/>");  
7     document.write("1 ==> 0001 <br/>");  
8     document.write("2 ==> 0010 <br/>");  
9     document.write("3 ==> 0011 <br/>");  
10    document.write("4 ==> 0000 <br/>");  
11    document.write("5 ==> 0001 <br/>");  
12    document.write("6 ==> 0010 <br/>");  
13    document.write("7 ==> 0011 <br/>");  
14    document.write("8 ==> 0000 <br/>");  
15    document.write("9 ==> 0001 <br/>");  
16    document.write("10 => 0010 <br/>");  
17    document.write("11 => 0011 <br/>");  
18    document.write("12 => 0000 <br/>");  
19    document.write("13 => 0001 <br/>");  
20    document.write("14 => 0010 <br/>");  
21    document.write("15 => 0011 <br/>");  
22 }
```

Grafica 7. Código de la función completa del programa 3

El código completo se visualiza así:

```
DECIMAL A BINARIO (0 - 15)  
  
La combinación de números binarios debe ser diferente para cada número  
decimal. Esto garantiza que cada número es diferente de los demás.  
  
0 ==> 0000  
1 ==> 0001  
2 ==> 0010  
3 ==> 0011  
4 ==> 0000  
5 ==> 0001  
6 ==> 0010  
7 ==> 0011  
8 ==> 0000  
9 ==> 0001  
10 => 0010  
11 => 0011  
12 => 0000  
13 => 0001  
14 => 0010  
15 => 0011
```

Grafica 7. Código completa del programa 3

En el programa tres se utiliza la misma función solo que cambien que los números decimales ya pasan a binarios y se le agregan una nueva función que es **texto(cadena)**, que su principal tarea almacenar datos que se pueden representar en forma de texto.

```
function binarios_con_funcion() {  
  texto("NÚMEROS BINARIOS DEL 0 AL 15");  
  texto("");  
  texto("0 ==> 0000");  
  texto("1 ==> 0001");  
  texto("2 ==> 0010");  
  texto("3 ==> 0011");  
  texto("4 ==> 0100");  
  texto("5 ==> 0101");  
  texto("6 ==> 0110");  
  texto("7 ==> 0111");  
  texto("8 ==> 1000");  
  texto("9 ==> 1001");  
  texto("10 ==> 1010");  
  texto("11 ==> 1011");  
  texto("12 ==> 1100");  
  texto("13 ==> 1101");  
  texto("14 ==> 1110");  
  texto("15 ==> 1111");  
}  
  
function texto( cadena ) {  
  document.write( cadena + "<br/>" );  
}
```

Grafica 8. Código de la función completa incluyendo de la cadena del programa 4

Así se vería la página en con este código:

```
NÚMEROS BINARIOS DEL 0 AL 15  
  
0 ==> 0000  
1 ==> 0001  
2 ==> 0010  
3 ==> 0011  
4 ==> 0100  
5 ==> 0101  
6 ==> 0110  
7 ==> 0111  
8 ==> 1000  
9 ==> 1001  
10 ==> 1010  
11 ==> 1011  
12 ==> 1100  
13 ==> 1101  
14 ==> 1110  
15 ==> 1111
```

Grafica 9. Código completo del programa 4

En el quinto y último código se agrega una función nueva que es el “switch”, que esta es una condicional, que a diferencia de “if”, no necesariamente tiene que cumplir con una condición, simplemente compara una variable “case”, en esta variable no hay necesidad de demostrar un valor verdadero y falso, simplemente hacer operaciones para un número indeterminado de valores. Dentro de la variable “case” hace la declaración del valor buscado hasta encontrar con otra variable “break”, que ejecuta por completa esta función.

Se determina una nueva variable “var i=0;”, y la condición “while (i < 16) { mostrar_binario(i); i = i +1;}” que determina el numero binarios de todos los números menos de 16, con la función “switch” se determina los posible números con los “cases” y su completa ejecución con “break”. Su código completo se presentaría así:

```
1  function binarios_con_switch() {
2      texto("NÚMEROS BINARIOS DEL 0 AL 15 CON SWITCH");
3      texto("");
4
5      var i = 0;
6      while (i < 16) {
7          mostrar_binario( i );
8          i = i + 1;
9      }
10 }
11
12 function mostrar_binario( numero ) {
13     switch ( numero ) {
14         case 0:
15             texto("0 ==> 0000");
16             break;
17         case 1:
18             texto("1 ==> 0001");
19             break;
20         case 2:
21             texto("2 ==> 0010");
22             break;
23         case 3:
24             texto("3 ==> 0011");
25             break;
26         case 4:
27             texto("4 ==> 0100");
28             break;
29         case 5:
```

Grafica 10. Código completo del programa 5

```
29     case 5:  
30         texto("5 ==> 0101");  
31         break;  
32     case 6:  
33         texto("6 ==> 0110");  
34         break;  
35     case 7:  
36         texto("7 ==> 0111");  
37         break;  
38     case 8:  
39         texto("8 ==> 1000");  
40         break;  
41     case 9:  
42         texto("9 ==> 1001");  
43         break;  
44     case 10:  
45         texto("10 => 1010");  
46         break;  
47     case 11:  
48         texto("11 => 1011");  
49         break;  
50     case 12:  
51         texto("12 => 1100");  
52         break;  
53     case 13:  
54         texto("13 => 1101");  
55         break;
```

Grafica 11. Código completo del programa 5

```
56     case 14:  
57         texto("14 => 1110");  
58         break;  
59     case 15:  
60         texto("15 => 1111");  
61         break;  
62     default:  
63         texto("-----");  
64         break;  
65     }  
66 }  
67  
68 function texto( cadena ) {  
69     document.write( cadena + "<br/>" );  
70 }
```

Grafica 12. Código completo del programa 5

Así sería la visualización con el programa completo:

NÚMEROS BINARIOS DEL 0 AL 15 CON SWITCH

```
0 ==> 0000
1 ==> 0001
2 ==> 0010
3 ==> 0011
4 ==> 0100
5 ==> 0101
6 ==> 0110
7 ==> 0111
8 ==> 1000
9 ==> 1001
10 ==> 1010
11 ==> 1011
12 ==> 1100
13 ==> 1101
14 ==> 1110
15 ==> 1111
```

Grafica 13. Código del programa 5

4 BIBLIOGRAFÍA

<https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Sentencias/switch>

<https://desarrolloweb.com/articulos/estructura-case-vbscript.html>

<https://developer.mozilla.org/es/docs/Web/HTML/Elemento/br>

<https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Sentencias/do...while>

<https://www.elvisualista.com/2016/10/20/que-son-los-numeros-binarios/>