

---

# Disco Codificador

**Gonzalo Mena Mendoza**

*mena@computer.org*

*http://mena.com.mx/gonzalo/maestria/algor/DiscoCodificador.nb*

*Proyecto de la materia de Análisis de Algoritmos y Estructuras de Datos Avanzadas  
Profesor: Dr. Jaime Rangel Mondragón*

*Facultad de Informática  
Universidad Autónoma de Querétaro  
abril de 2005*

Dada una serie de  $2^n$  cadenas binarias n bits, dibujar el disco codificador de pistas concéntricas correspondiente.

## Referencias

[1] Donald E. Knuth. "Generating All Possibilities", *The Art of Computer Programming, Volumen 4, prefascículo 2a*, [en línea], [versión del 31 de marzo de 2005]

<<http://www-cs-faculty.stanford.edu/~knuth/news.html>>

<<http://www-cs-faculty.stanford.edu/~knuth/fasc2a.ps.gz>>

[2] Gonzalo Mena Mendoza. "Código Binario Gray", [en línea], [consulta 20 de abril de 2005]

<<http://mena.com.mx/gonzalo/maestria/algor/CodigoBinarioGray.nb>>

---

## Introducción

Se puede identificar la posición de un disco rotatorio que ha sido dividido en  $2^n$  partes utilizando n sensores, cada uno de los cuales tiene que distinguir entre el blanco y el negro de la pista concéntrica correspondiente.

La siguiente figura muestra el resultado deseado. Del lado izquierdo ese encuentra el disco para el orden lexicográfico de  $n = 4$ ; y del lado derecho, el disco del código binario Gray respectivo[1].

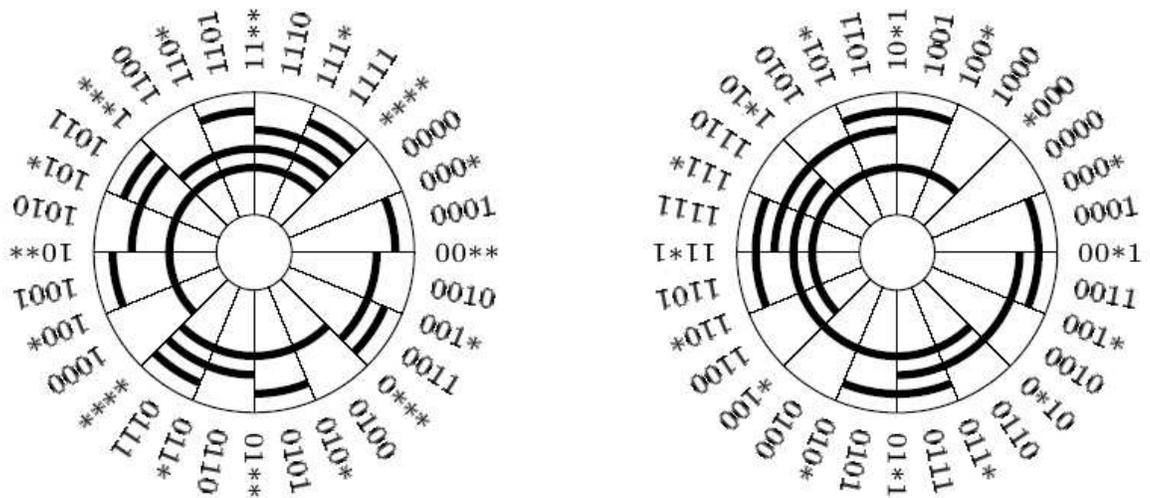


Figure 1

## Análisis

La construcción del disco codificador puede dividirse en las siguientes etapas:

1. Generación de las series binarias
2. Construcción de las pistas concéntricas
3. Separación de las pistas
4. Separación de los sectores
5. Generación de las etiquetas radiales.

### ■ Series Binarias

#### ■ Orden Lexicográfico

```
OrdenLexicográfico[n_] := IntegerDigits[Range[0, 2n - 1], 2, n]
```

```
OrdenLexicográfico[3]
```

```
{ {0, 0, 0}, {0, 0, 1}, {0, 1, 0}, {0, 1, 1},  
  {1, 0, 0}, {1, 0, 1}, {1, 1, 0}, {1, 1, 1} }
```

#### ■ Código Gray

Se puede definir el Código Gray de la siguiente manera recursiva [2]:

```

CódigoGray[0] := {}
CódigoGray[n_] := Join[
  Map[Prepend[#, 0] &, CódigoGray[n - 1]],
  Map[Prepend[#, 1] &, Reverse[CódigoGray[n - 1]]]
]

CódigoGray[3]

{{0, 0, 0}, {0, 0, 1}, {0, 1, 1}, {0, 1, 0},
 {1, 1, 0}, {1, 1, 1}, {1, 0, 1}, {1, 0, 0}}

```

## ■ Pistas

El problema principal consiste en dibujar las pistas codificadoras concéntricas. El disco se divide en  $2^n$  sectores, uno por elemento de la serie  $s$ . Para cada sector se dibujan  $n$  secciones de pista, cada una de las secciones corresponde a un dígito del elemento en cuestión. El dígito más significativo corresponde a la pista más interna. El 1 se representa con negro y el 0 con blanco. Para hacer más distingible la pista más significativa, se pueden comenzar a dibujar las pistas a partir de un radio  $r$ .

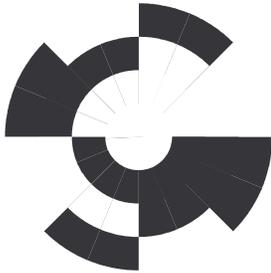
```

Pistas[s_, n_, r_] := {
  Table[ (* un sector por elemento *)
    Table[ (* una sección de pista por dígito *)
      {
        GrayLevel[1 - s[[i]][[j]], (* 0 : blanco, 1 : negro *)
        Disk[{0, 0}, j + r, { $\frac{2\pi}{2^n} (i - 1)$ ,  $\frac{2\pi}{2^n} i$ }]
      },
      {j, n, 1, -1} (* dibujar de afuera hacia adentro *)
    ],
    {i, 1, 2^n}
  ],
  {GrayLevel[1], Disk[{0, 0}, r]} (* disco interior vacío *)
}

Pistas[s_] := Pistas[s,  $\frac{1}{\text{Log}[\text{Length}[s], 2]}$ , 1]

```

```
Pistas[OrdenLexicográfico[3]];
Show[Graphics[%], AspectRatio → Automatic]
```



- Graphics -

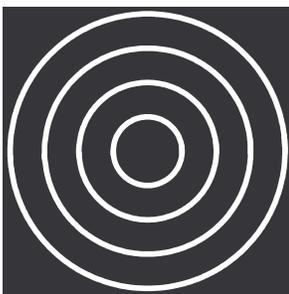
## ■ Aros Separadores

Es conveniente separar las pistas concéntricas para hacerlas más visibles.

```
Aros[n_, r_] := {GrayLevel[1], Thickness[0.02],
  Table[Circle[{0, 0}, i + r], {i, 0, n}]
}
Aros[n_] := Aros[n, 1]

Aros[3]
Show[Graphics[%], AspectRatio → Automatic, Background → GrayLevel[0]]

{GrayLevel[1], Thickness[0.02], {Circle[{0, 0}, 1],
  Circle[{0, 0}, 2], Circle[{0, 0}, 3], Circle[{0, 0}, 4]}}
```



- Graphics -

## ■ Radios

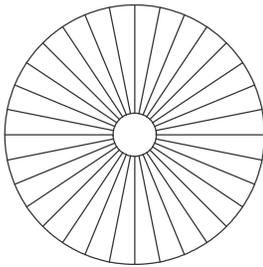
Así mismo, es conveniente separar los sectores.

```

Radios[n_, r_] := {
  Table[
    Line[{{r Cos[ $\frac{2 \pi}{2^n} i$ ], r Sin[ $\frac{2 \pi}{2^n} i$ ]},
      {(n + r) Cos[ $\frac{2 \pi}{2^n} i$ ], (n + r) Sin[ $\frac{2 \pi}{2^n} i$ ]}]],
    {i, 1, 2n}
  ],
  Circle[{0, 0}, r], Circle[{0, 0}, n + r]
  (* círculo interior y exterior *)
}
Radios[n_] := Radios[n, 1]

Radios[5];
Show[Graphics[%], AspectRatio -> Automatic]

```



- Graphics -

## ■ Etiquetas

### ■ Texto Radial

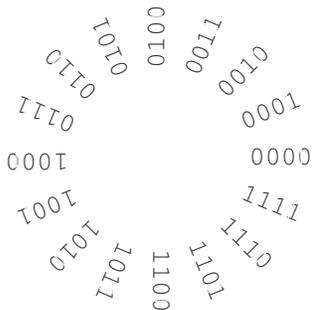
Se requiere de una función que dibuje todos los elementos de una serie de manera radial.

```

Etiquetas[s_, m_, r_] := {
  Table[
    Text[
      TableForm[s[[i + 1]], TableDirections -> {Row, Column},
        TableSpacing -> 0] (* sin llaves *),
      {r Cos[ $\frac{2 \pi}{m} i$ ], r Sin[ $\frac{2 \pi}{m} i$ ]}, (* posición *)
      {0, 0}, (* centrado *)
      {Cos[ $\frac{2 \pi}{m} i$ ], Sin[ $\frac{2 \pi}{m} i$ ]} (* rotación *)
    ],
    {i, 0, m - 1}
  ],
  {GrayLevel[1], Circle[{0, 0}, r + 1]} (* círculo exterior
    para forzar que se dibujen las etiquetas completas *)
}
Etiquetas[s_] := Etiquetas[s, Length[s], Length[s[[1]]] + 2]

Etiquetas[OrdenLexicográfico[4]];
Show[Graphics[%], AspectRatio -> Automatic]

```



- Graphics -

### ■ Diferencias entre elementos contiguos

Se quiere que para cada sector no sólo se muestre el valor numérico del elemento sino que entre sectores se muestren los dígitos que son diferentes entre elemento contiguos.

```

MostrarDiferencias[s_] :=
  Flatten[
    Table[ (* para cada elemento de la serie *)
      {
        Flatten[
          Table[ (* para cada dígito del elemento *)
            {If[ (* poner un asterisco si el
                  dígito es diferente al del elemento anterior *)
              s[[i]][[j]] == RotateRight[s][[i]][[j]],
              s[[i]][[j]], "*"
            ]},
            {j, 1, Length[s[[1]]]}
          ]
        ],
        s[[i]] (* poner el elemento en cuestión *)
      },
      {i, 1, Length[s]}
    ],
    1
  ]

OrdenLexicográfico[3]
MostrarDiferencias[%]

{{0, 0, 0}, {0, 0, 1}, {0, 1, 0}, {0, 1, 1},
 {1, 0, 0}, {1, 0, 1}, {1, 1, 0}, {1, 1, 1}}

{(*, *, *), {0, 0, 0}, {0, 0, *}, {0, 0, 1}, {0, *, *},
 {0, 1, 0}, {0, 1, *}, {0, 1, 1}, {*, *, *}, {1, 0, 0},
 {1, 0, *}, {1, 0, 1}, {1, *, *}, {1, 1, 0}, {1, 1, *}, {1, 1, 1}}

```

## ■ Disco Codificador

La construcción del disco codificador se reduce a reunir las funciones anteriores.

```

DiscoCodificador[s_, n_] :=
  {
    Pistas[s],
    Aros[n],
    Radios[n],
    Etiquetas[MostrarDiferencias[s]]
  }

DiscoCodificador[s_] := DiscoCodificador[s,  $\frac{1}{\text{Log}[\text{Length}[s], 2]}$ ]

```

## Resultado

```

n = 4;
Show[
GraphicsArray[{
Graphics[
DiscoCodificador[OrdenLexicografico[n]],
AspectRatio -> Automatic,
TextStyle -> {FontSize -> 9}, PlotLabel -> "Orden Lexicografico"
],
Graphics[
DiscoCodificador[CodigoGray[n]],
AspectRatio -> Automatic,
TextStyle -> {FontSize -> 9}, PlotLabel -> "Codigo Gray"
]
}
]];

```

