

# Taller 2: Representación booleana a partir de un número natural\*

Frank Hernández<sup>a</sup>, Juliana Bejarano<sup>1</sup>

<sup>a</sup>*Pontificia Universidad Javeriana, Bogotá, Colombia*

---

## Abstract

En este documento se presenta el problema de escribir un algoritmo para solucionar el problema de calcular la representación booleana de un número natural usando la estrategia de "dividir y vencer".

*Keywords:* algoritmo, escritura formal, dividir y vencer, booleano, número natural.

---

## 1. Análisis del problema

La definición de número natural y su representación booleana, se representa como:

$$S = \langle S_i : i \in \{1..N\} \subset \mathbb{N} \rangle$$

Y se representa como  $\mathbb{B}$ .

Dónde:

- $S = \langle s_i \mid a_i \in \mathbb{N} \rangle$ , es la secuencia que contiene los elementos.
- $s_i \in S$ , es la posición de cada elemento.

## 2. Diseño del problema

El análisis anterior nos permite diseñar el problema: definir las entradas y salidas de un posible algoritmo de solución, que aún no está definido.

1. Entradas:
  - a)  $S = \langle s_i \mid a_i \in \mathbb{N} \rangle$ , la secuencia de elementos.
  - b)  $b$ , punto de partida de la secuencia.
  - c)  $e$ , última posición de la secuencia.
2. Salidas:  $B$ , la representación booleana del elemento.

---

\*Este documento presenta la escritura formal de un algoritmo.

*Email addresses:* `fs.hernandez1@javeriana.edu.co` (Frank Hernández),  
`juliana_bejarano@javeriana.edu.co` (Juliana Bejarano)

### 3. Algoritmos de solución

#### 3.1. Algoritmo recursivo

Este algoritmo de solución es una traducción literal de las deficiones de lo que se quiere resolver.

---

**Algorithm 1** Representación booleana de un número natural.

---

```

1: procedure BOOLEAN_REP( $s$ )
2:   if  $s = 0$  then
3:     return 0
4:   end if
5:   if  $s = 1$  then
6:     return 1
7:   end if
8:    $cantbits \leftarrow \text{CANTIDAD}(s)$ 
9:   return BOOLEAN_REP_AUX( $s, 0, cantbits - 1$ )
10: end procedure
11: procedure CANTIDAD( $s$ )
12:    $cantbits \leftarrow 0$ 
13:   while  $s \gg cantbits$  do
14:      $cantbits \leftarrow cantbits + 1$ 
15:   end while
16:   return  $cantbits$ 
17: end procedure
18: procedure BOOLEAN_REP_AUX( $s, b, e$ )
19:   if  $b = e$  then
20:     return  $\text{str}((s \gg b) \& 1)$ 
21:   end if
22:    $q \leftarrow (b + e) // 2$ 
23:    $L \leftarrow \text{BOOLEAN\_REP\_AUX}(s, b, q)$ 
24:    $R \leftarrow \text{BOOLEAN\_REP\_AUX}(s, q + 1, e)$ 
25:   return  $R + L$ 
26: end procedure

```

---

##### 3.1.1. Invariante

La invariante del algoritmo es  $s$ , el número que se recibe, dado que:

- **Inicio:** Al comienzo, la invariante se asegura de que la función está preparada para construir la representación binaria correcta de todos los bits de  $s$ .
- **Avance:** En cada paso recursivo, la invariante se mantiene porque cada subproblema se resuelve correctamente para su respectivo intervalo de bits, y estos se combinan para formar la representación binaria.

- **Terminación:** Cuando el caso base se alcanza y la recursión termina, la invariante asegura que todos los bits han sido procesados correctamente, y la representación binaria completa de  $s$  se ha generado.

### 3.1.2. Análisis de complejidad

El algoritmo `BOOLEAN_REP` tiene orden de complejidad  $O(2T(n/2) + O(1))$ , calculado a partir del teorema maestro debido a que no hay ciclos, y hay 2 llamados recursivos. Al resolver la ecuación por teorema maestro, llegamos a la complejidad  $O(s)$ .

## 4. Plan de pruebas

### 4.0.1. Numero: 0

1. resultado esperado: 0
2. resultado Obtenido:

```
Ingrese el número: 0
La representación booleana del número es 0
```

Figura 1: Resultado numero 0

### 4.0.2. Numero: 3

1. resultado esperado: 11
2. resultado Obtenido:

```
Ingrese el número: 3
La representación booleana del número es 11
```

Figura 2: Resultado numero 3

### 4.0.3. Numero: 28

1. resultado esperado: 11100
2. resultado Obtenido:

```
Ingrese el número: 28
La representación booleana del número es 11100
```

Figura 3: Resultado numero 28

4.0.4. *Numero: 126*

1. resultado esperado: 1111110
2. resultado Obtenido:

```
Ingrese el número: 126
La representación booleana del número es 1111110
```

Figura 4: Resultado numero 126

4.0.5. *Numero: 1002*

1. resultado esperado: 1111101010
2. resultado Obtenido:

```
Ingrese el número: 1002
La representación booleana del número es 1111101010
```

Figura 5: Resultado numero 1002