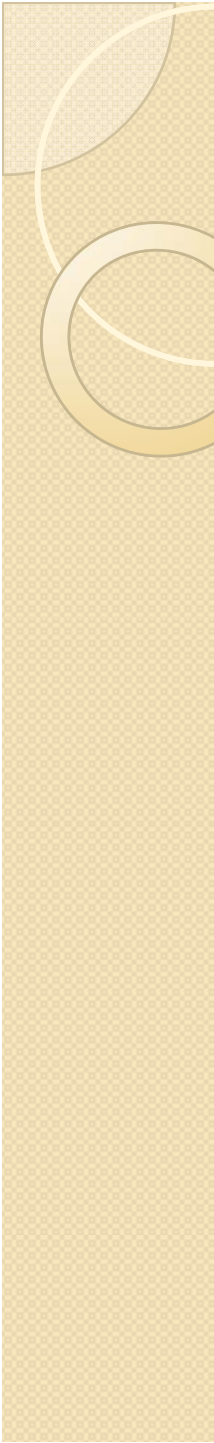


# Conversion de Decimales positivos a binario

---

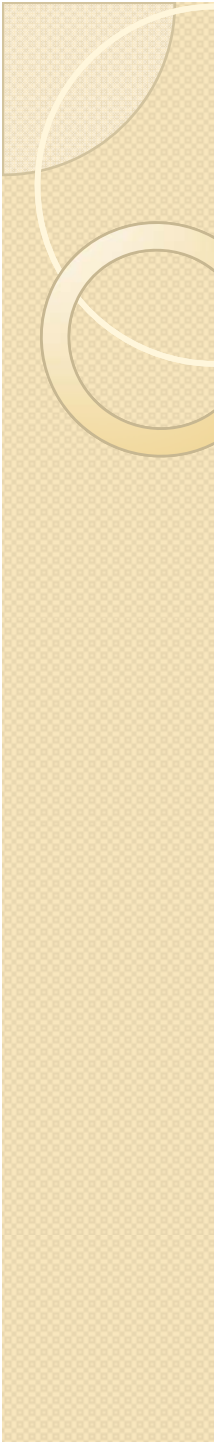
Cesar A Aceros Moreno

INGE3016



# Objetivos

- Reforzar la razon por la que el analisis y el diseno son importantes.
- Diseñar 2 aplicaciones usando estructuras de repeticion para convertir de decimal a binario.
  - Sencilla usando outputs por cada bit.
  - Usando strings para guardar la respuesta.
- Mejorar destrezas de codificacion en VBA.



# Conceptos Requeridos

- Basicos:

- Saber convertir de decimal a binario.
- Manejar Top-Down Development Program
- Saber escribir flujogramas.
- Saber hacer pruebas manuales.
- Codificar programas en VBA.

- Avanzados:

- Definir que es un string.
- Saber unir 2 strings (Operaciones con Strings).

# Conversion de Decimal a Binario (Revisada)

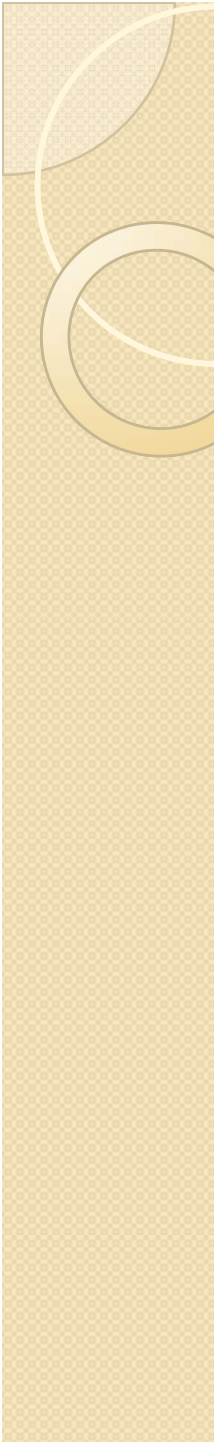
| Numero | Residuo | Cociente |
|--------|---------|----------|
| 14     | 0       | 7        |
| 7      | 1       | 3        |
| 3      | 1       | 1        |
| 1      | 1       | 0        |
| 0      | 0       | 0        |
| 0      | 0       | 0        |
| 0      | 0       | 0        |
| 0      | 0       | 0        |

| Numero | Residuo | Cociente |
|--------|---------|----------|
| 65     | 1       | 32       |
| 32     | 0       | 16       |
| 16     | 0       | 8        |
| 8      | 0       | 4        |
| 4      | 0       | 2        |
| 2      | 0       | 1        |
| 1      | 1       | 0        |
| 0      | 0       | 0        |

Cierto que cuando el cociente llega a 0 ha terminado la conversion?  
seguro que si porque de ahi en adelante solo hay 0s.

# Ahora una mas larga....

| Numero | Residuo | Cociente |
|--------|---------|----------|
| 1254   | 0       | 627      |
| 627    | 1       | 313      |
| 313    | 1       | 156      |
| 156    | 0       | 78       |
| 78     | 0       | 39       |
| 39     | 1       | 19       |
| 19     | 1       | 9        |
| 9      | 1       | 4        |
| 4      | 0       | 2        |
| 2      | 0       | 1        |
| 1      | 1       | 0        |
| 0      | 0       | 0        |



## Primer Diseno:

Este diseno es uno que hace la conversion sin mucha complicacion.

El objetivo es que usted entienda como se hace para poner en un while que valida que cuando el cociente en la conversion de decimal a binario llega a 0 es hora de terminar la conversion.

En el siguiente slide usted puede ver como se hace el proceso detallado de diseno.

# TUTORIA SOBRE REPETICIONES:

①

EL PROPOSITO DE ESTA TUTORIA ES REVISAR 2 FORMAS DE DISEÑAR UN PROGRAMA PARA CONVERTIR DECIMALES ENTEROS POSITIVOS A NUMEROS BINARIOS.

## PRIMER DISEÑO: (FORMA SIMPLE)

COMO ES UN DISEÑO UNO DEBE DEFINIR EL MARCO AL CUAL SE AJUSTA LO QUE UNO ESTA DISEÑANDO.

EN ESTE CASO VOY A DECIR QUE:

DISEÑO: PROGRAMA QUE CONVIERTE NUMEROS DECIMALES ENTEROS POSITIVOS A NUMEROS BINARIOS. EL PROGRAMA REALIZARA LA CONVERSION REPETIDAMENTE Y LA UNICA FORMA DE TERMINAR EL PROGRAMA SERA COLOCANDO UN NUMERO MENOR QUE CERO.

## VARIABLES:

NUM: ENTERO (DEFINIDO POR USUARIO)  
NUMERO DECIMAL QUE SE VA A CONVERTIR.

bit: ENTERO (USADO POR EL PROG).  
NUMERO QUE ALMACENA EL BIT QUE SE ESTA CONVIRTIENDO.

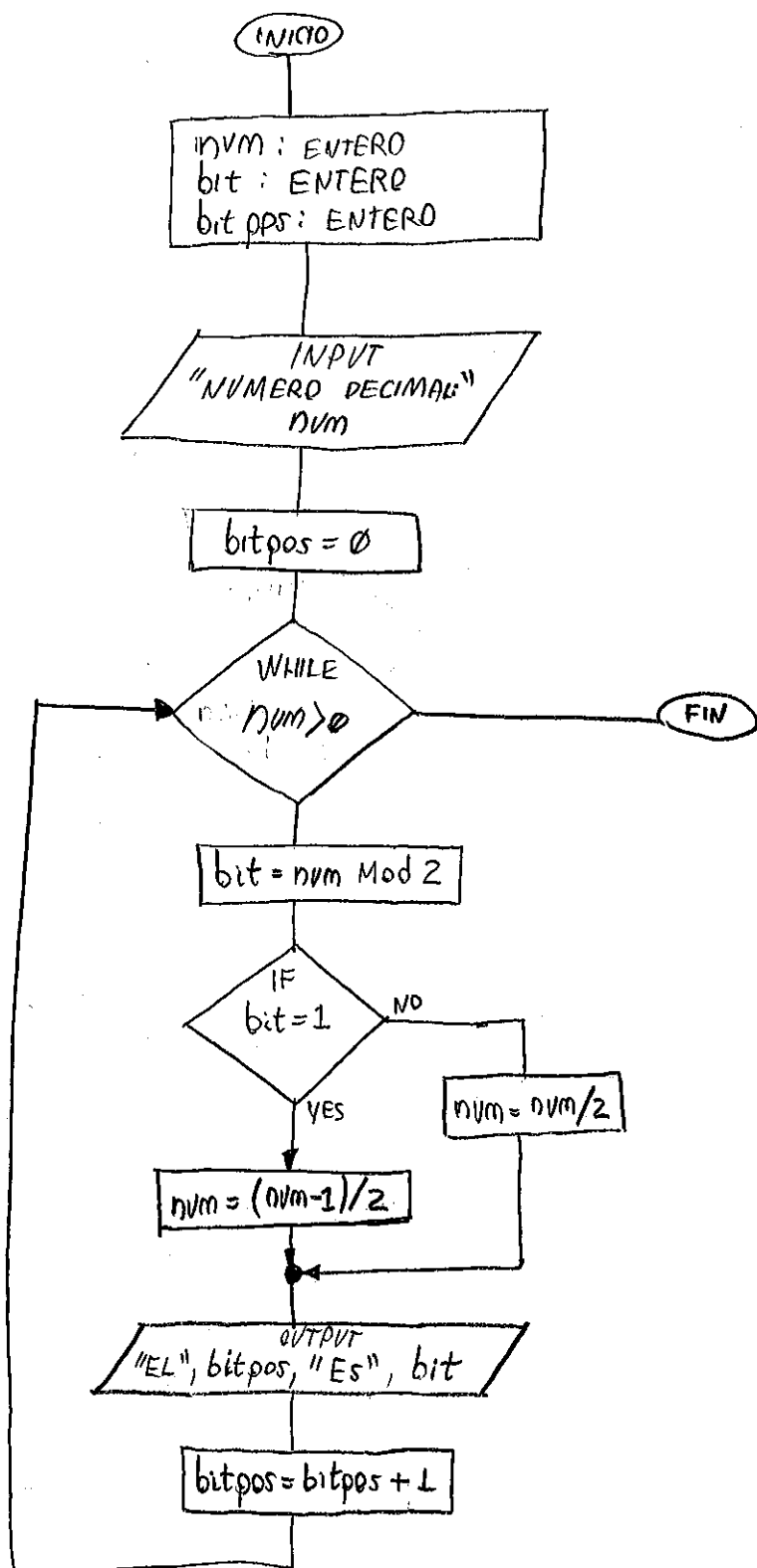
bitpos: ENTERO (USADO POR EL PROG)  
NUMERO QUE ALMACENA EL NUMERO DEL BIT QUE SE ESTA CONVIRTIENDO.

ENTRADA DE NUM.

mientras que  $NUM > 0$  hago  
convierte num a binario.

# FLUJOGRAMA.

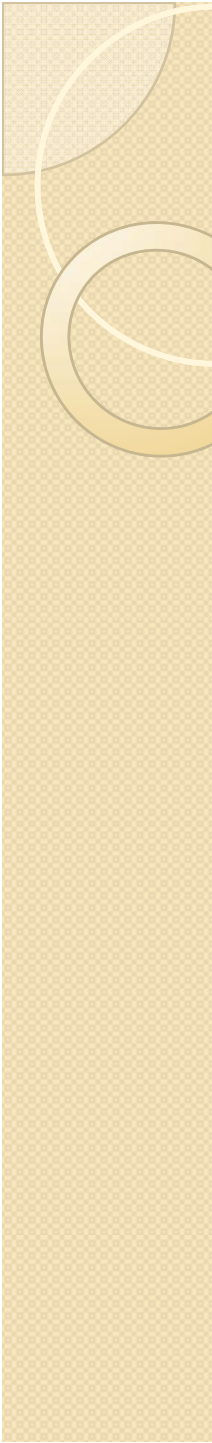
2



3) PRUEBA MANUAL:

| RUND | NUM                                      | bit          | bitpos | OUTPUT       |
|------|------------------------------------------|--------------|--------|--------------|
|      | NUM=10<br>10 > 0 TRUE                    | 10 mod 2 = 0 | 0      |              |
|      | NUM = $\frac{10}{2} = 5$<br>5 > 0 TRUE   | 5 MOD 2 = 1  | 1      | → bit 0 ES 0 |
|      | NUM = $\frac{5-1}{2} = 2$<br>2 > 0 TRUE  | 2 MOD 2 = 0  | 2      | → bit 1 ES 1 |
|      | NUM = $\frac{2}{2} = 1$<br>1 > 0 TRUE    | 1 MOD 2 = 1  | 3      | → bit 2 ES 0 |
|      | NUM = $\frac{1-1}{2} = 0$<br>0 > 0 FALSE |              | 4      | → bit 3 ES 0 |
|      |                                          |              |        | → END.       |





Antes del Segundo Diseno:

HABLEMOS DE....

---

STRINGS

Que son?

# Que es un String?

**Datos Alfanuméricos (String):** Es una secuencia de caracteres alfanuméricos que permiten representar valores identificables de forma descriptiva, esto incluye nombres de personas, direcciones, etc. Este tipo de datos se representan encerrados entre comillas.

Ejemplos:

|              |   |   |   |   |   |   |   |   |   |     |
|--------------|---|---|---|---|---|---|---|---|---|-----|
| "Maria"      | = | M | a | r | i | a |   |   |   |     |
| "15432.23"   | = | 1 | 5 | 4 | 3 | 2 | . | 2 | 3 |     |
| "124.32E-12" | = | 1 | 2 | 4 | . | 3 | 2 | E | - | 1 2 |
| "10110101"   | = | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |     |

# Que es un caracter alfanumerico?

|    |    |    |    |   |    |   |     |   |     |   |     |   |     |   |     |   |     |   |     |   |
|----|----|----|----|---|----|---|-----|---|-----|---|-----|---|-----|---|-----|---|-----|---|-----|---|
| 0  | 24 | †  | 48 | 0 | 72 | H | 96  |   | 120 | x | 144 | E | 168 | ¿ | 192 | L | 216 | ± | 240 | ≡ |
| 1  | 25 | ↓  | 49 | 1 | 73 | I | 97  | a | 121 | y | 145 | æ | 169 | ƒ | 193 | ⊥ | 217 | ‡ | 241 | ± |
| 2  | 26 | +  | 50 | 2 | 74 | J | 98  | b | 122 | z | 146 | æ | 170 | ŕ | 194 | ⊥ | 218 | ‡ | 242 | ± |
| 3  | 27 | +  | 51 | 3 | 75 | K | 99  | c | 123 | ( | 147 | ö | 171 | ŕ | 195 | ⊥ | 219 | ‡ | 243 | ± |
| 4  | 28 | ⊥  | 52 | 4 | 76 | L | 100 | d | 124 | l | 148 | ö | 172 | ŕ | 196 | ⊥ | 220 | ‡ | 244 | ± |
| 5  | 29 | ⊥  | 53 | 5 | 77 | M | 101 | e | 125 | ) | 149 | ö | 173 | ŕ | 197 | ⊥ | 221 | ‡ | 245 | ± |
| 6  | 30 | ⊥  | 54 | 6 | 78 | N | 102 | f | 126 | ~ | 150 | ü | 174 | ŕ | 198 | ⊥ | 222 | ‡ | 246 | ± |
| 7  | 31 | ⊥  | 55 | 7 | 79 | O | 103 | g | 127 | Δ | 151 | ü | 175 | ŕ | 199 | ⊥ | 223 | ‡ | 247 | ± |
| 8  | 32 |    | 56 | 8 | 80 | P | 104 | h | 128 | Ç | 152 | ü | 176 | ŕ | 200 | ⊥ | 224 | α | 248 | ° |
| 9  | 33 | !  | 57 | 9 | 81 | Q | 105 | i | 129 | Ü | 153 | ö | 177 | ŕ | 201 | ⊥ | 225 | β | 249 | ° |
| 10 | 34 | "  | 58 | : | 82 | R | 106 | j | 130 | é | 154 | ü | 178 | ŕ | 202 | ⊥ | 226 | Γ | 250 | ° |
| 11 | 35 | #  | 59 | : | 83 | S | 107 | k | 131 | ä | 155 | ç | 179 | ŕ | 203 | ⊥ | 227 | Π | 251 | √ |
| 12 | 36 | \$ | 60 | < | 84 | T | 108 | l | 132 | ä | 156 | ç | 180 | ŕ | 204 | ⊥ | 228 | Σ | 252 | ° |
| 13 | 37 | ℥  | 61 | = | 85 | U | 109 | m | 133 | ö | 157 | ŕ | 181 | ŕ | 205 | ⊥ | 229 | σ | 253 | ° |
| 14 | 38 | £  | 62 | > | 86 | V | 110 | n | 134 | ä | 158 | ç | 182 | ŕ | 206 | ⊥ | 230 | μ | 254 | ° |
| 15 | 39 | ·  | 63 | ? | 87 | W | 111 | o | 135 | ç | 159 | ç | 183 | ŕ | 207 | ⊥ | 231 | γ | 255 | ° |
| 16 | 40 | (  | 64 | @ | 88 | X | 112 | p | 136 | ö | 160 | ä | 184 | ŕ | 208 | ⊥ | 232 | δ |     |   |
| 17 | 41 | )  | 65 | À | 89 | Y | 113 | q | 137 | ö | 161 | í | 185 | ŕ | 209 | ⊥ | 233 | θ |     |   |
| 18 | 42 | ×  | 66 | B | 90 | Z | 114 | r | 138 | ö | 162 | ó | 186 | ŕ | 210 | ⊥ | 234 | Ω |     |   |
| 19 | 43 | +  | 67 | C | 91 | [ | 115 | s | 139 | í | 163 | ú | 187 | ŕ | 211 | ⊥ | 235 | δ |     |   |
| 20 | 44 | ,  | 68 | D | 92 | \ | 116 | t | 140 | í | 164 | ñ | 188 | ŕ | 212 | ⊥ | 236 | ω |     |   |
| 21 | 45 | -  | 69 | E | 93 | ] | 117 | u | 141 | í | 165 | ñ | 189 | ŕ | 213 | ⊥ | 237 | φ |     |   |
| 22 | 46 | .  | 70 | F | 94 | ^ | 118 | v | 142 | Ä | 166 | ä | 190 | ŕ | 214 | ⊥ | 238 | € |     |   |
| 23 | 47 | /  | 71 | G | 95 | _ | 119 | w | 143 | Ä | 167 | ä | 191 | ŕ | 215 | ⊥ | 239 | Ω |     |   |

# Como se veria "Maria" en ASCII

Las computadoras entienden solamente lenguaje binario. Por lo cual todo lo que se procesa en 0's y 1's. La forma como los caracteres son interpretados por una computadora es a traves de los valores binarios de cada caracter.

| String |   |   |   |   |   |   |   |  |  |   | Codificacion ASCII |    |     |     |    |    |    |    |  |  |
|--------|---|---|---|---|---|---|---|--|--|---|--------------------|----|-----|-----|----|----|----|----|--|--|
| M      | a | r | i | a |   |   |   |  |  | = | 77                 | 97 | 114 | 105 | 97 |    |    |    |  |  |
| I      | 0 | I | I | 0 | I | 0 | I |  |  | = | 49                 | 48 | 49  | 49  | 48 | 49 | 48 | 49 |  |  |

NOTA:

Un archivo de una computadora es una secuencia de caracteres.

# Operaciones con Strings

**Asignación:** Consiste en asignarle una cadena a otra.

A="Maria"

**Concatenación:** Consiste en unir dos cadenas o más (o una cadena con un carácter) para formar una cadena de mayor tamaño.

A="1011"      y    B = "10"

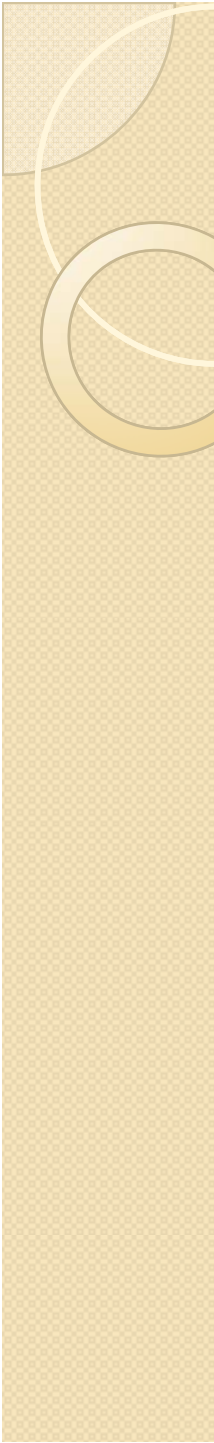
En VBA:            A & B = "101110"

**Búsqueda:** Consiste en localizar dentro de una cadena una subcadena más pequeña o un carácter.

**Extracción:** Se trata de sacar fuera de una cadena una porción de la misma según su posición dentro de ella.

**Comparación:** Se utiliza para comparar dos cadenas.

Fuente: <http://es.wikipedia.org/wiki/String>



## Segundo Diseno:

Este diseno pretende validar mas cosas que el anterior:

- Si pones numeros negativos da error.
- Para salir lo haces con (-1)
- Funciona para convertir 0.

Pretende que usted desarrolle mas su estilo de programacion.

SEGUNDO DISEÑO: ESTE ES MAS ROBUSTO QUE EL ANTERIOR. (5)

DISEÑO: PROGRAMA QUE CONVIERTE NUMEROS DECIMALES ENTEROS POSITIVOS A NUMEROS BINARIOS. EL PROGRAMA REALIZARA LA CONVERSION REPETIDAMENTE CON -1 PARA SALIR Y TODA ENTRADA NEGATIVA DISTINTA DE -1 SERA CONSIDERADA UN ERROR.

VARIABLES:

num: NUMERO A CONVERTIR (ENTERO) DADO POR USUARIO.

numa: VARIABLE QUE ALMACENA UNA COPIA DE NUM. (PROGRAMA) (ENTERO)

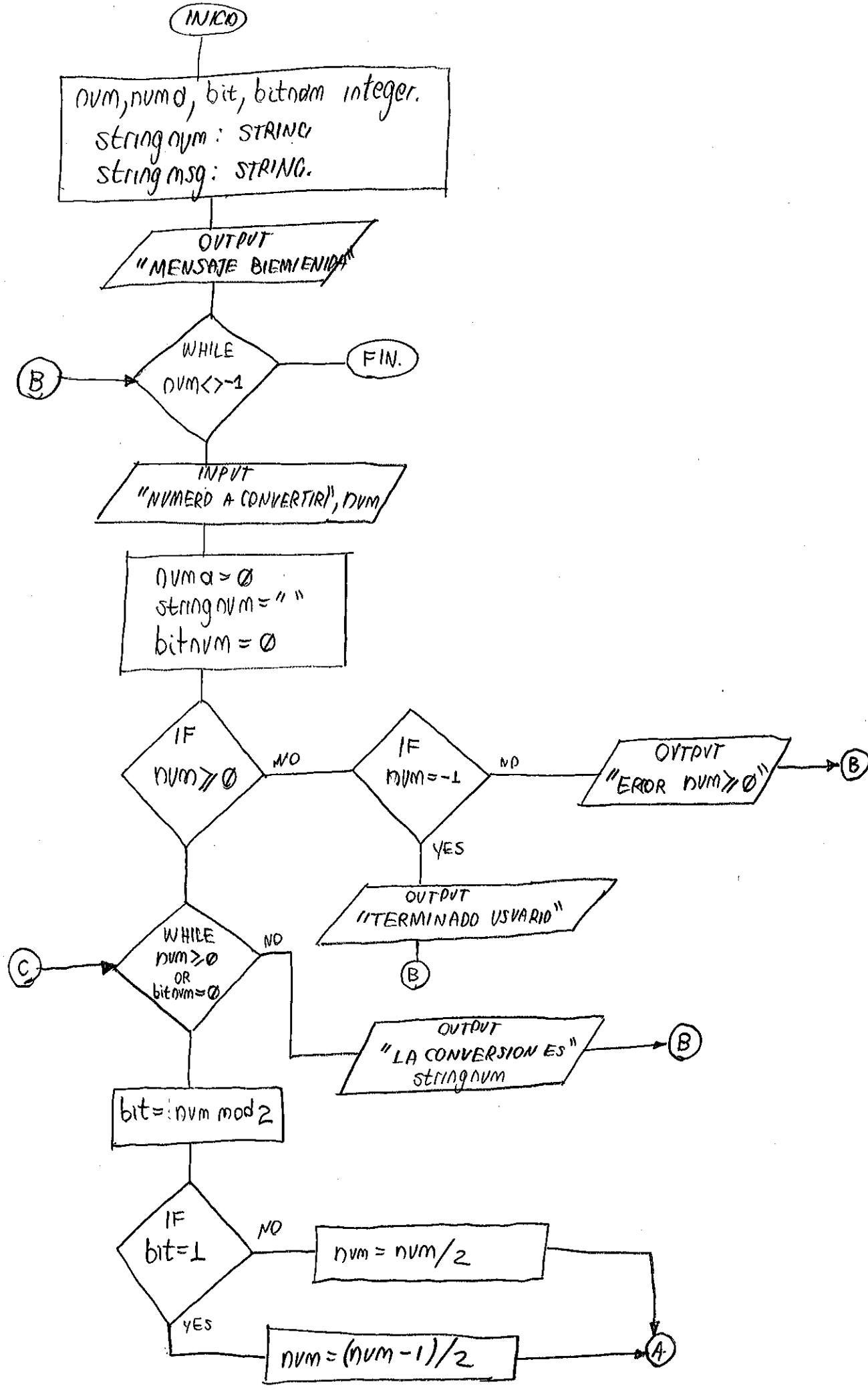
bit: VARIABLE QUE TIENE EL BIT QUE SE ESTA CONVIRTIENDO. ENTERO

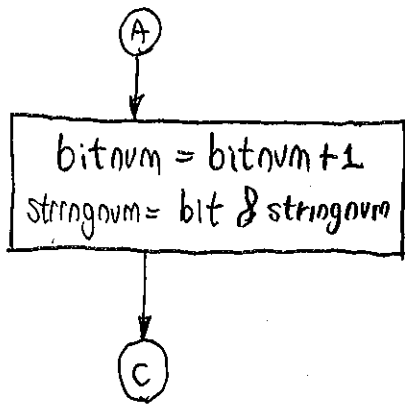
stringnum: VARIABLE QUE CONTIENE TODO EL NUMERO BINARIO.  
(STRING)

NOTA!!

ES IMPORTANTE QUE REVISE LA PRESENTACION QUE ESTA EN LA PAGINA DEL CURSO. SI NO SABE QUE ES UN STRING.

LA PRUEBA MANUAL LA DEJO COMO EJERCICIO.





LO RETO A QUE TRATE DE HACER QUE EL PROGRAMA FALLE  
NO ES DIFÍCIL HACERLO...

SÓLO PIENSE CUAL ES EL NÚMERO  
EN EL CUAL EMPIEZA A FALLAR.

DE CUANTOS BITS ES ESE NÚMERO?