



Facultad de Ingeniería

Laboratorios de computación salas A y B

Profesor:

Adrian Ulises Mercado Martínez

Asignatura:

Fundamentos de programación

07

Grupo:

No de Proyecto(s):

1

Integrante(s):

Arredondo Granados Gerardo
Casillas Herrera Leonardo Didier
Diaz Gonzalez Rivas Angel Iñaqui
Galván Castro Martha Selene

*No. de Equipo de cómputo
empleado:*

20

No. de Lista o Brigada:

04

Semestre:

2022-1

Fecha de entrega:

18 de Diciembre del 2021

Observaciones:

CALIFICACIÓN: _____

Proyecto 1

Conversión de números

Introducción	4
Desarrollo	5
Conversiones de binario a decimal	5
Algoritmo	5
Pruebas de escritorio	5
pseudocódigo:	6
Diagrama de flujo	7
Pruebas de escritorio	8
Código	8
Conversiones de decimal a binario	9
Algoritmo	9
Pruebas de escritorio	9
Pseudocódigo	10
Diagrama de flujo	11
Pruebas de escritorio	11
Código	12
Conversión de hexadecimal a decimal	13
Algoritmo	13
Pruebas de escritorio	13
Pseudocódigo	14
Diagrama de flujo	15
Pruebas de escritorio	15
Código	16
Conversiones de decimal a hexadecimal	18
Algoritmo	18
Pruebas de escritorio	18
Pseudocódigo	19
Diagrama de flujo	19
Pruebas de escritorio	20
Código	20
Conversiones de octal a decimal	22
Algoritmo	22
Pruebas de escritorio	22
Pseudocódigo	23
Diagrama de flujo	23
Pruebas de escritorio	24
Código	24
Conversiones de decimal a octal	26
Algoritmo	26
Pruebas de escritorio	26

Pseudocódigo	27
Diagrama de flujo	28
Pruebas de escritorio	28
Código	29
Conversiones de octal a binario	30
Algoritmo	30
Pruebas de escritorio	30
Pseudocódigo	31
Diagrama de flujo	32
Pruebas de escritorio	34
Código	34
Conversiones de Binario a Octal	36
Algoritmo	36
Pruebas de escritorio	36
Pseudocódigo	37
Diagrama de flujo	38
Pruebas de escritorio	39
Código	39
Conversiones de Hexadecimal a Binario	41
Algoritmo	41
Pruebas de escritorio	41
Pseudocódigo	42
Diagrama de flujo	44
Pruebas de escritorio	45
Código	46
Conversiones de Binario a Hexadecimal	48
Algoritmo	48
Pruebas de escritorio	48
Pseudocódigo	49
Diagrama de flujo	51
Pruebas de escritorio	52
Código	52
Manual de usuario	55
Referencias	56

Introducción

En el siguiente proyecto se realizó un programa en lenguaje C capaz de realizar conversiones entre numérica de los sistema de numérico decimal, binario, octal y hexadecimal los cuales son: decimal a binario, binario a decimal, decimal a octal, octal a decimal, decimal a hexadecimal, hexadecimal a decimal, binario a hexadecimal, hexadecimal a binario, octal a binario y binario a octal

Desarrollo

Conversiones de binario a decimal

Algoritmo

Para convertir de **binario** a **decimal** se siguen los siguientes pasos:

1. Solicitar un número binario y almacenarlo en una variable
2. Acomodar los números del último al primero.
3. Reescribir las potencias de 2 solo en los números donde el número binario sea igual a 1 y en los que el número binario sea 0 se escribe 0.
4. Sumar los números en donde su valor sea uno.
5. Colocar el resultado de la suma.

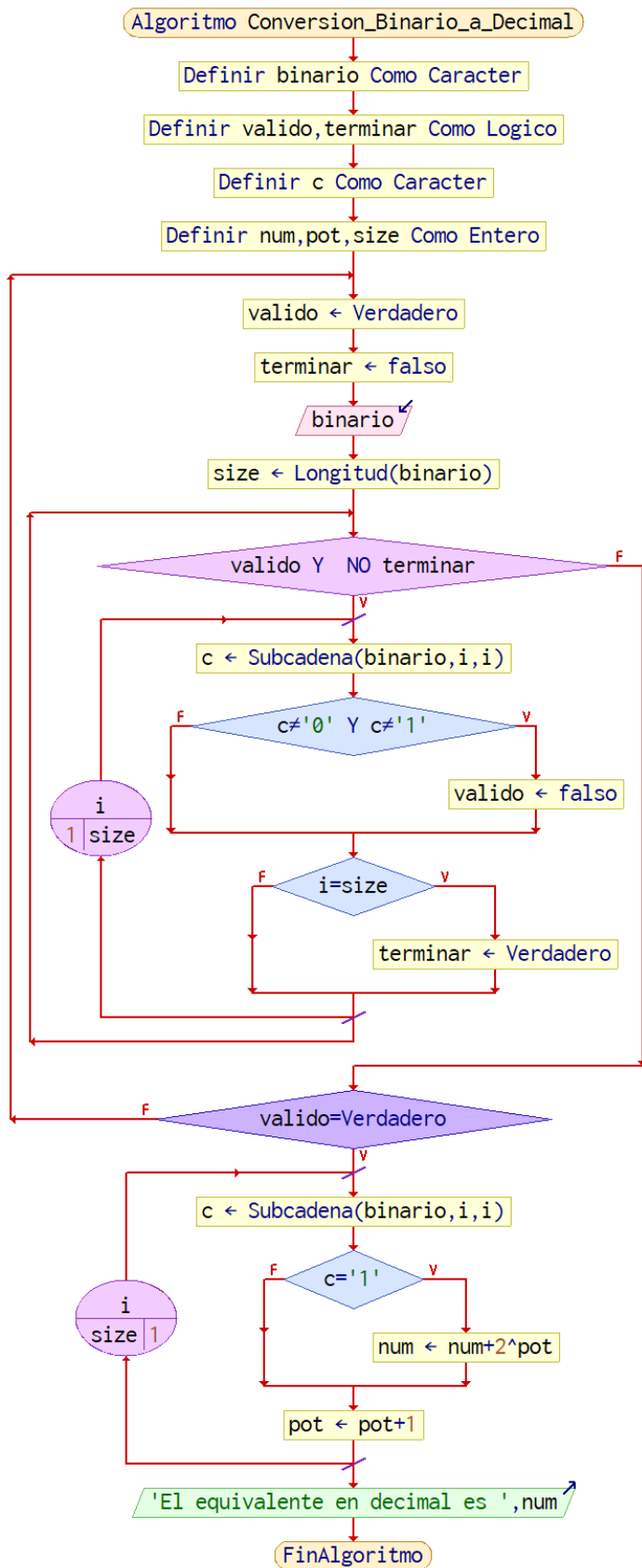
Pruebas de escritorio

Iteración	Binario		Decimal
1	10010	01001 020016 2+16=Decimal	18
2	1110110	0 1 1 0 1 1 1 0 2 4 0 16 32 64 2+4+16+32+64=Decimal	118

pseudocódigo:

```
1  Algoritmo Conversion_Binario_a_Decimal
2      Definir binario Como Cadena
3      Definir valido, terminar Como Logico
4      Definir c Como Caracter
5      Definir num, pot, size Como Entero
6
7      Repetir
8          valido  $\leftarrow$  Verdadero
9          terminar  $\leftarrow$  Falso
10         Leer binario
11         size  $\leftarrow$  Longitud(binario)
12         Mientras valido y no terminar Hacer
13             Para i $\leftarrow$ 1 Hasta size Hacer
14                 c  $\leftarrow$  Subcadena(binario, i,i)
15                 Si c $\neq$ "0" y c  $\neq$  "1" Entonces
16                     valido  $\leftarrow$  falso
17                 FinSi
18                 Si i = size Entonces
19                     terminar = Verdadero
20                 FinSi
21             FinPara
22         FinMientras
23     Hasta Que valido = Verdadero
24
25     Para i $\leftarrow$ size Hasta 1 Con Paso 1 Hacer
26         c  $\leftarrow$  Subcadena(binario, i,i)
27         Si c = "1" Entonces
28             num  $\leftarrow$  num + 2pot
29         FinSi
30         pot  $\leftarrow$  pot + 1
31     FinPara
32     Escribir "El equivalente en decimal es ", num
33 FinAlgoritmo
```

Diagrama de flujo



Pruebas de escritorio

```
PSelnt - Ejecutando proceso BINARIO_A_DECIMAL
*** Ejecución Iniciada. ***
Escriba el número a convertir
> 1111011
El equivalente a decimal es:123
*** Ejecución Finalizada. ***
```

Conversiones de decimal a binario

Algoritmo

Para convertir de **decimal** a **binario** se sigue los siguientes pasos:

- 1) Solicitar un número decimal y almacenarlo en una variable
- 2) Dividir el decimal entre 2.
- 3) Si el resultado es par se asigna como valor 0, y si es impar el 1.
- 4) Continuamos los 2 pasos anteriores hasta llegar al resultado 0.
- 5) Los ceros y unos obtenidos se deben anotar de arriba hacia abajo.
- 6) Acomodando los residuos (mod) orden inverso lo que se obtiene es el decimal convertido a binario.

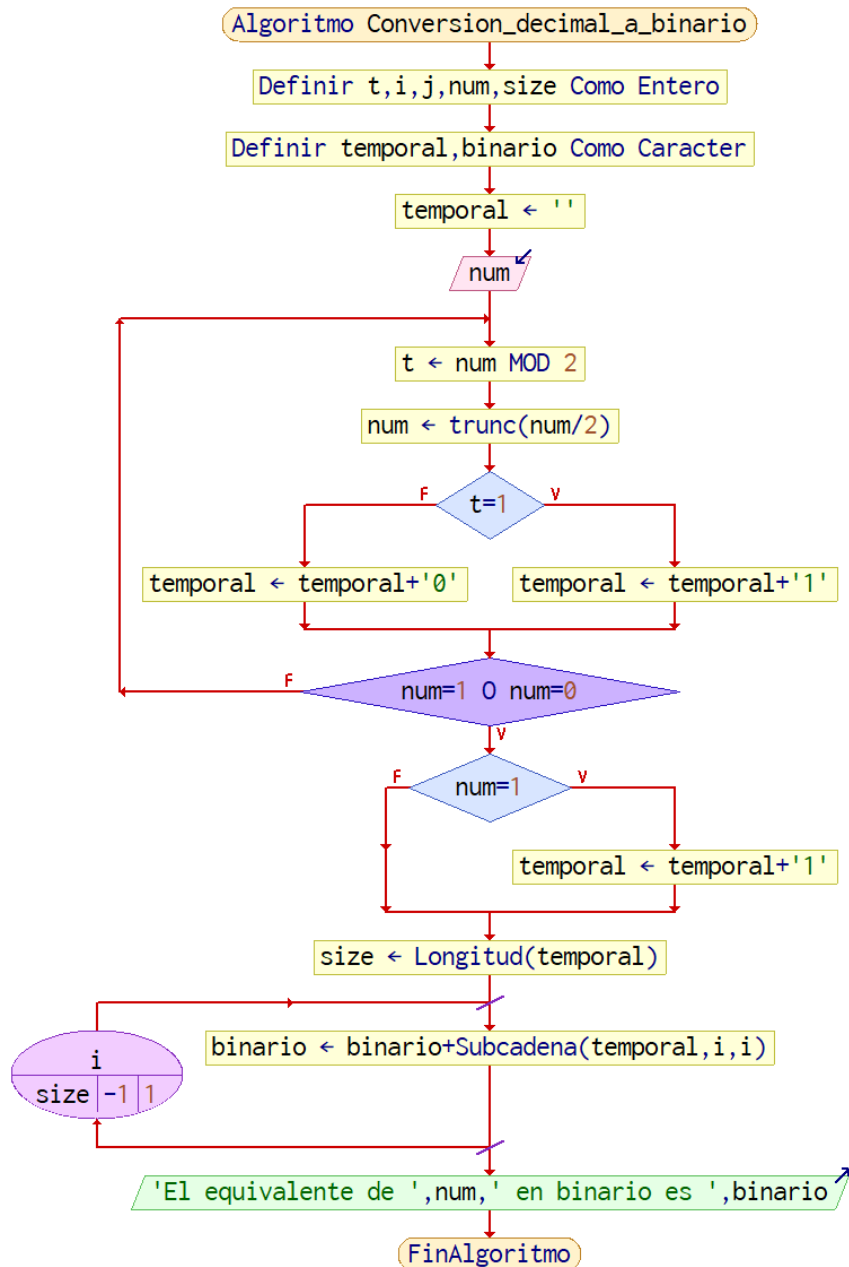
Pruebas de escritorio

Iteración	Decimal		Binario
1	30	30/2=15 par 15/2=1 impar 7/2=1 impar 3/2=1 impar 1/2= 1 impar	11110
2	18	18/2 =9 par 9/2= 4 impar 4/2=2 par 2/2= 1 impar	1010

Pseudocódigo

```
1  Algoritmo Conversion_decimal_a_binario
2      Definir t, i, j, num, size Como Entero
3      Definir temporal, binario Como Caracter
4      temporal  $\leftarrow$  ""
5      Leer num
6      Repetir
7          t  $\leftarrow$  num mod 2
8          num  $\leftarrow$  trunc(num/2) // Obteniendo la división entera
9          Si t= 1 Entonces
10             temporal $\leftarrow$ temporal+"1"
11          SiNo
12             temporal $\leftarrow$  temporal + "0"
13          FinSi
14      Hasta Que num = 1 o num= 0
15      Si num = 1 Entonces
16          temporal $\leftarrow$ temporal + "1"
17      FinSi
18      size  $\leftarrow$  Longitud(temporal)
19      Para i $\leftarrow$  size Hasta 1 Con Paso -1 Hacer
20          binario  $\leftarrow$  binario + Subcadena(temporal, i, i)
21      FinPara
22      Escribir "El equivalente de ", num " en binario es ", binario
23
24
25  FinAlgoritmo
```

Diagrama de flujo



Pruebas de escritorio

```
PSInt - Ejecutando proceso CONVERSION_DECIMAL_A_B...
*** Ejecución Iniciada. ***
Escriba el número a convertir
> 445
El equivalente de 445 a binario es es:0000000110111101
*** Ejecución Finalizada. ***
```

Conversión de hexadecimal a decimal

Algoritmo

Para convertir de **hexadecimal** a **decimal** se sigue los siguientes pasos:

1. Solicitar un número decimal y almacenarlo en una variable .
2. Dividir el número entre 16
3. Si el cociente es mayor o igual que 16, se divide entre 16.
4. Se continúa dividiendo entre 16 hasta obtener un cociente que sea menor que 16, de no ser así se regresa al punto 3.
5. Una vez teniendo el cociente se escribirán los siguientes valores en el orden especificado: Último cociente, Último resto, Penúltimo resto y seguir los valores hasta llegar al Primer resto.
6. Los valores de 0, 1, 2, 3, 4, 5, 6, 7, 8 y 9 no se modifican, los siguientes valores se van a cambiar: el 10 por el A, el 11 por el B, el 12 por el C, el 13 por el D, el 14 por el E y el 15 por el F.
7. Como resultado tendríamos un número en base hexadecimal.

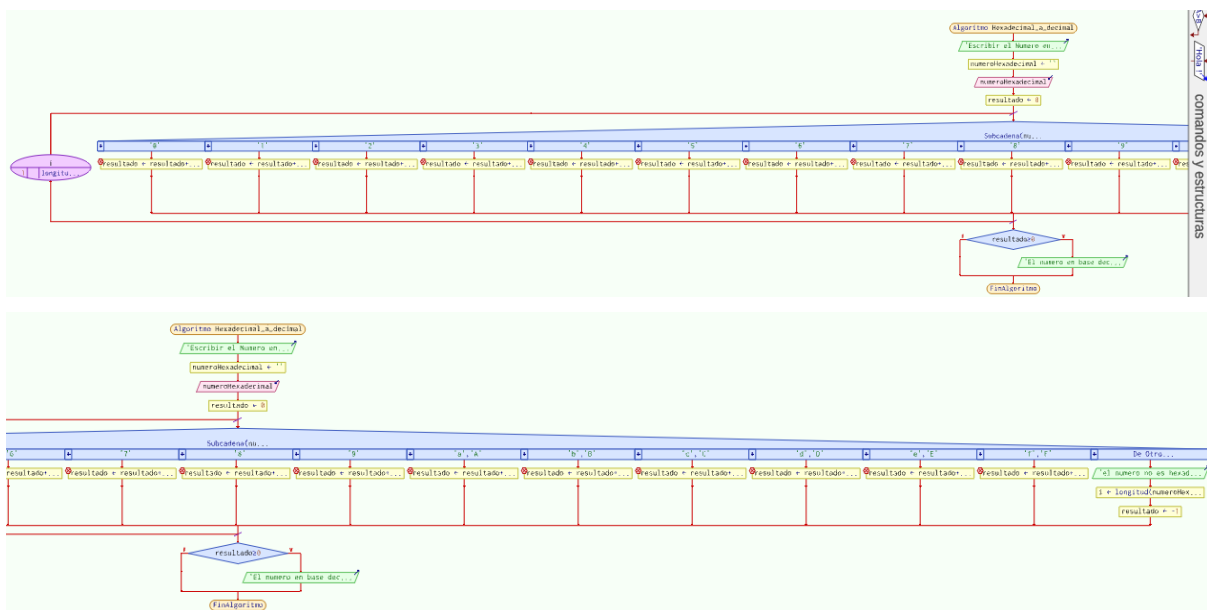
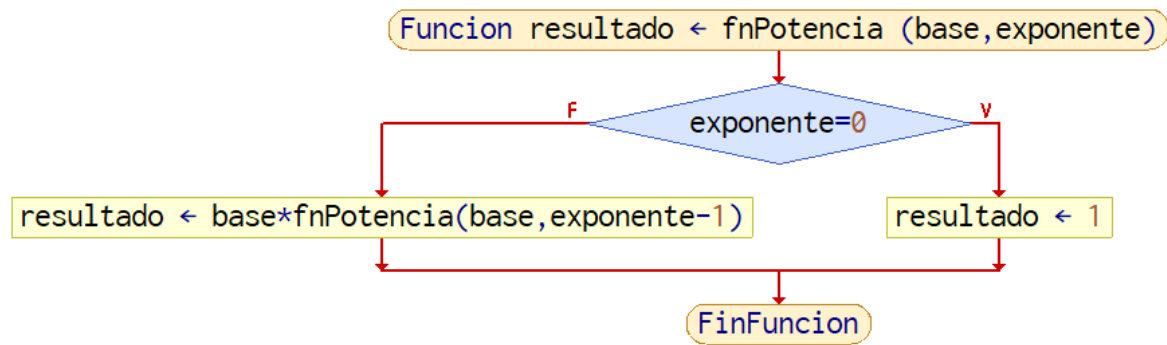
Pruebas de escritorio

Iteración	Hexadecimal		Decimal
1	6B	B=12 6 11 6 $11 \cdot 16^0$ $6 \cdot 16^1$ 11 96 $11 + 96 = 108$	108
2	91A	A=10 9 $10 \cdot 16^0$ $9 \cdot 16^1$ 10 144 $10 + 144 = 154$	154

Pseudocódigo

```
1  SubProceso resultado ← fnPotencia (base, exponente)
2  Si exponente=0 Entonces
3  | resultado ← 1
4  SiNo
5  | resultado ← base*fnPotencia(base,exponente-1)
6  FinSi
7  FinSubProceso
8  Algoritmo Hexadecimal_a_decimal
9  | Escribir "Escribir el Numero en hexadecimal"
10 numeroHexadecimal ← ""
11 leer numeroHexadecimal
12 resultado← 0
13 Para i←1 Hasta Longitud(numeroHexadecimal)
14 | Segun Subcadena(numeroHexadecimal,i,i) Hacer
15 |   '0':
16 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
17 |   '1':
18 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
19 |   '2':
20 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
21 |   '3':
22 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
23 |   '4':
24 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
25 |   '5':
26 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
27 |   '6':
28 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
29 |   '7':
30 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
31 |   '8':
32 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
33 |   '9':
34 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
35 |   'a', 'A':
36 |     resultado = resultado +10 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
37 |   'b', 'B':
38 |     resultado = resultado +11 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
39 |   'c', 'C':
40 |     resultado = resultado +12 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
41 |   'd', 'D':
42 |     resultado = resultado +13 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
43 |   'e', 'E':
44 |     resultado = resultado +14 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
45 |   'f', 'F':
46 |     resultado = resultado +15 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
47 |   De Otro Modo:
48 |     Escribir "el numero no es hexadecimal"
49 |     i← longitud(numeroHexadecimal)
50 |     resultado=-1
51 |   Fin Segun
52 |   Fin Para
32 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
33 |   '9':
34 |     resultado = resultado + ConvertirANumero(Subcadena(numeroHexadecimal,i,i)) * fnPotencia(16,Longitud(numeroHexadecimal)-i)
35 |   'a', 'A':
36 |     resultado = resultado +10 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
37 |   'b', 'B':
38 |     resultado = resultado +11 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
39 |   'c', 'C':
40 |     resultado = resultado +12 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
41 |   'd', 'D':
42 |     resultado = resultado +13 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
43 |   'e', 'E':
44 |     resultado = resultado +14 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
45 |   'f', 'F':
46 |     resultado = resultado +15 * fnPotencia(16,Longitud(numeroHexadecimal)-i)
47 |   De Otro Modo:
48 |     Escribir "el numero no es hexadecimal"
49 |     i← longitud(numeroHexadecimal)
50 |     resultado=-1
51 |   Fin Segun
52 |   Fin Para
53 si resultado ≥ 0
54 |   Escribir "El numero en base decimal es", resultado
55 FinSi
56 FinAlgoritmo
```

Diagrama de flujo



Pruebas de escritorio

```
PSInt - Ejecutando proceso CONVERSION_HEXADECIMAL_A_DECIMAL

*** Ejecución Iniciada. ***
Escriba el número en hexadecimal
> 34B
El número en base decimal es:843
*** Ejecución Finalizada. ***
```

Conversiones de decimal a hexadecimal

Algoritmo

Para convertir de **decimal** a **hexadecimal** se sigue los siguientes pasos:

- 1) Solicitar un número en base hexadecimal y almacenarlo en una variable.
- 2) Del el número en base hexadecimal, de derecha a izquierda, iremos multiplicando cada cifra por $1(16^0)$, la segunda cifra por 16 , la tercera por 16^2 , la cuarta por 16^3 , y continuaremos multiplicando las cifras por 16 y seguiremos aumentando la potencia de 16 por cada cifra que multipliquemos, hasta que no quede ninguna cifra.
- 3) Los resultados de las multiplicaciones se sumarán.
- 4) El resultado de la suma será un número decimal.

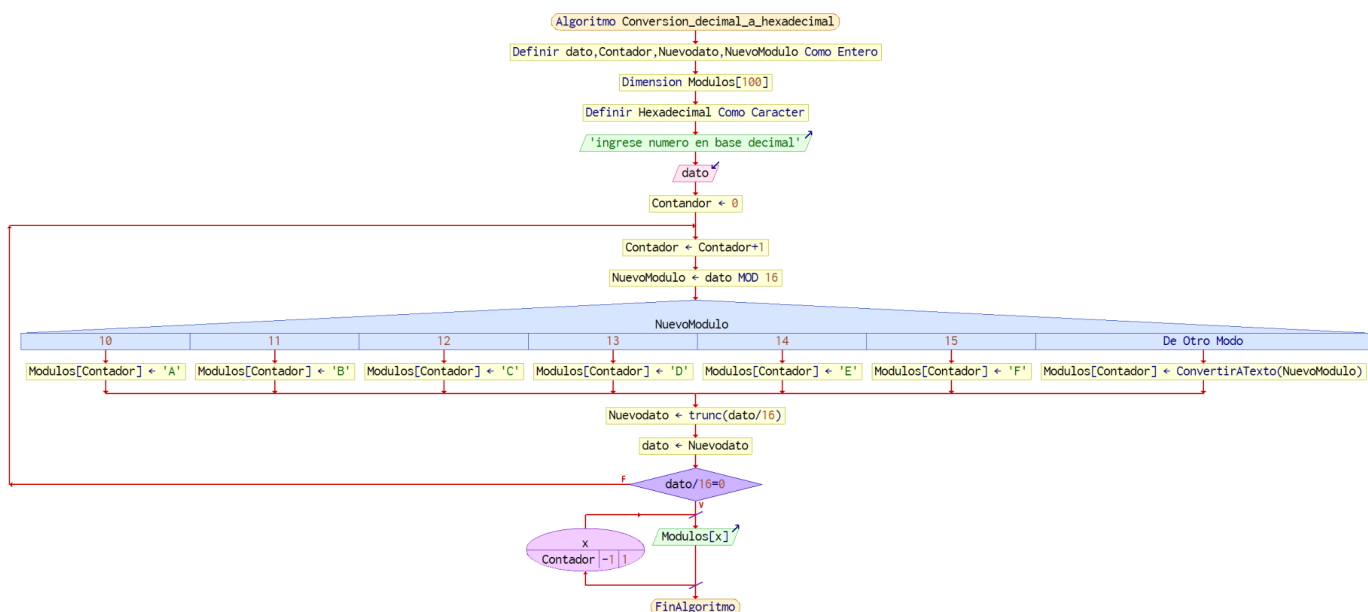
Pruebas de escritorio

Iteración	Decimal		Hexadecimal
1	103	130/16=8 residuo= 2 8/16= 0 residuo=8 82	82
2	280	280/16= 17 residuo=8 17/16= 1 residuo 1 1/16=0 residuo 1 118	118

Pseudocódigo

```
1  Algoritmo Conversion_decimal_a_hexadecimal
2      Definir dato, contador, Nuevodata, NuevoModulo Como Entero
3      Dimension Modulos[100]
4      Definir Hexadecimal Como Caracter
5      Imprimir "ingrese numero en base decimal"
6      Leer dato
7      Contador=0
8      Repetir
9          contador=contador+1
10         NuevoModulo= dato mod 16
11         Segun NuevoModulo Hacer
12             caso 10:
13                 Modulos[contador]="A"
14             Caso 11:
15                 Modulos[contador]="B"
16             Caso 12:
17                 Modulos[contador]="C"
18             Caso 13:
19                 Modulos[contador]="D"
20             Caso 14:
21                 Modulos[contador]="E"
22             Caso 15:
23                 Modulos[contador]="F"
24             De Otro Modo:
25                 Modulos[Contador]=ConvertirATexto(NuevoModulo)
26         Fin Segun
27         Nuevodata=trunc(dato/16)
28         dato=Nuevodata
29     Hasta Que dato/16=0
30     Para x  $\leftarrow$  Contador Hasta 1 con paso -1 Hacer
31         Escribir Modulos[x]
32     FinPara
33
34 FinAlgoritmo
```

Diagrama de flujo



Pruebas de escritorio

```
PSeInt - Ejecutando proceso CONVERSION_HEXADECIMA...  
*** Ejecución Iniciada. ***  
Escriba el número en base decimal  
> 1325  
El número en base hexadecimal es:52D  
*** Ejecución Finalizada. ***
```


Conversiones de octal a decimal

Algoritmo

Para convertir de **octal** a **decimal** se sigue los siguientes pasos:

- 1) Solicitar un número en base octal y almacenarlo en una variable.
- 2) Del el número en base octal, de derecha a izquierda, iremos multiplicando cada cifra por $1(8^0)$, la segunda cifra por 8 , la tercera por 8^2 , la cuarta por 8^3 , y continuaremos multiplicando las cifras por 8 y seguiremos aumentando la potencia de 8 por cada cifra que multipliquemos, hasta que no quede ninguna cifra.
- 3) Los resultados de las multiplicaciones se sumarán.
- 4) El resultado de la suma será un número decimal.

Pruebas de escritorio

Iteración	Hexadecimal		Decimal
1	196	$1 \leq 8$, $9 \neq 8$, $6 \leq 8$	no es un número en base octal.
2	670	078 $0 * 1 = 0$ $7 * 8 = 56$ $6 * 64 = 384$ $0 + 56 + 384 = 440$	440

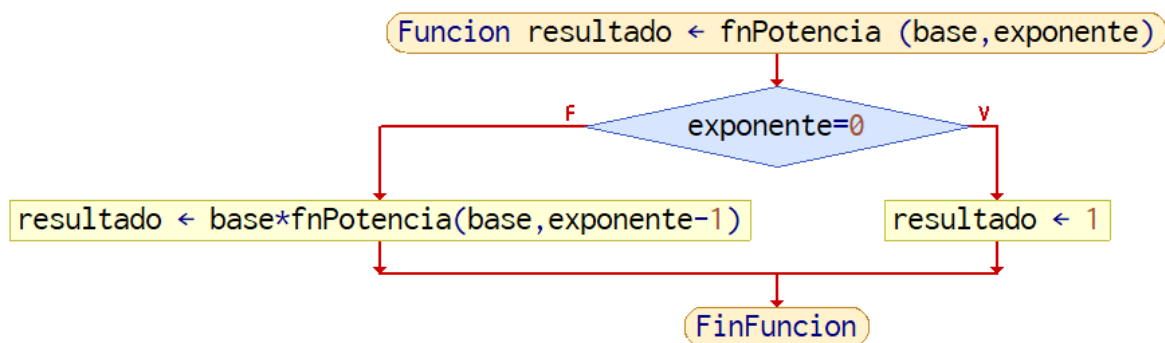
Pseudocódigo

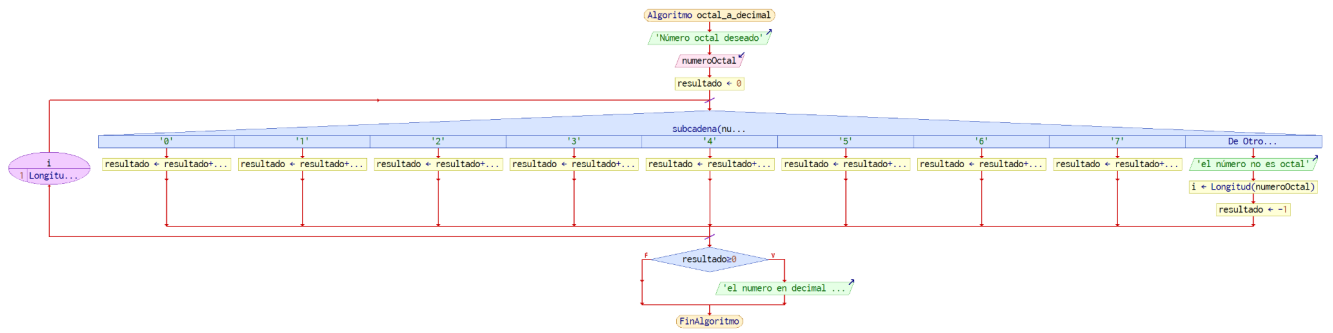
```

1  subproceso resultado ← fnPotencia (base, exponente)
2      si exponente=0 Entonces
3          resultado ← 1
4      SiNo
5          resultado ← base*fnPotencia(base, exponente-1)
6      FinSi
7  FinSubProceso
8
9  Algoritmo octal_a_decimal
10     escribir "Número octal deseado"
11     leer numeroOctal
12
13     resultado ← 0
14
15     para i←1 hasta longitud(numeroOctal)
16         segun Subcadena(numeroOctal,i,i) Hacer
17             "0":
18                 resultado = resultado + convertirANumero(subcadena(numeroOctal,i,i))*fnPotencia(8,Longitud(numeroOctal)-i)
19             "1":
20                 resultado = resultado + convertirANumero(subcadena(numeroOctal,i,i))*fnPotencia(8,Longitud(numeroOctal)-i)
21             "2":
22                 resultado = resultado + convertirANumero(subcadena(numeroOctal,i,i))*fnPotencia(8,Longitud(numeroOctal)-i)
23             "3":
24                 resultado = resultado + convertirANumero(subcadena(numeroOctal,i,i))*fnPotencia(8,Longitud(numeroOctal)-i)
25             "4":
26                 resultado = resultado + convertirANumero(subcadena(numeroOctal,i,i))*fnPotencia(8,Longitud(numeroOctal)-i)
27             "5":
28                 resultado = resultado + convertirANumero(subcadena(numeroOctal,i,i))*fnPotencia(8,Longitud(numeroOctal)-i)
29             "6":
30                 resultado = resultado + convertirANumero(subcadena(numeroOctal,i,i))*fnPotencia(8,Longitud(numeroOctal)-i)
31             "7":
32                 resultado = resultado + convertirANumero(subcadena(numeroOctal,i,i))*fnPotencia(8,Longitud(numeroOctal)-i)
33
34         De Otro Modo:
35             Escribir "el número no es octal"
36             i←Longitud(numeroOctal)
37             resultado = -1
38     FinSegun
39 FinPara
40
41 si resultado ≥ 0
42     Escribir "el numero en decimal es ",resultado
43 FinSi
44
45
46 FinAlgoritmo
47

```

Diagrama de flujo





Pruebas de escritorio

```

PSeInt - Ejecutando proceso CONVERSION_OCTAL_A_DEC...

*** Ejecución Iniciada. ***
Escriba el número en base octal
> 1071
El número en base decimal es:569
*** Ejecución Finalizada. ***
  
```

Conversiones de decimal a octal

Algoritmo

Para convertir de decimal a octal se sigue los siguientes pasos:

- 1) Solicitamos un número en base 8 y almacenamos la variable.
 - 1.1 verificar que el número sea base 8 sino así regresar al paso anterior.
- 2) Dividir el número entre 8.
- 3) Si el cociente es mayor o igual que 8, se divide entre 8.
- 4) Se continúa dividiendo entre 8 hasta obtener un cociente que sea menor que 8, de no ser así se regresa al punto 3.
- 5) Una vez teniendo el cociente se escribirán los siguientes valores en el orden especificado: Último cociente, Último resto, Penúltimo resto y seguir los valores hasta llegar al Primer resto.
- 6) Escribir los valores en el orden especificado en el punto 5, los valores deberían ser 0, 1, 2, 3, 4, 5, 6, y 7. Si entre los valores se encuentra el 8 o el 9, regrese al punto 3.
- 7) Como resultado tendríamos un número en base 8.

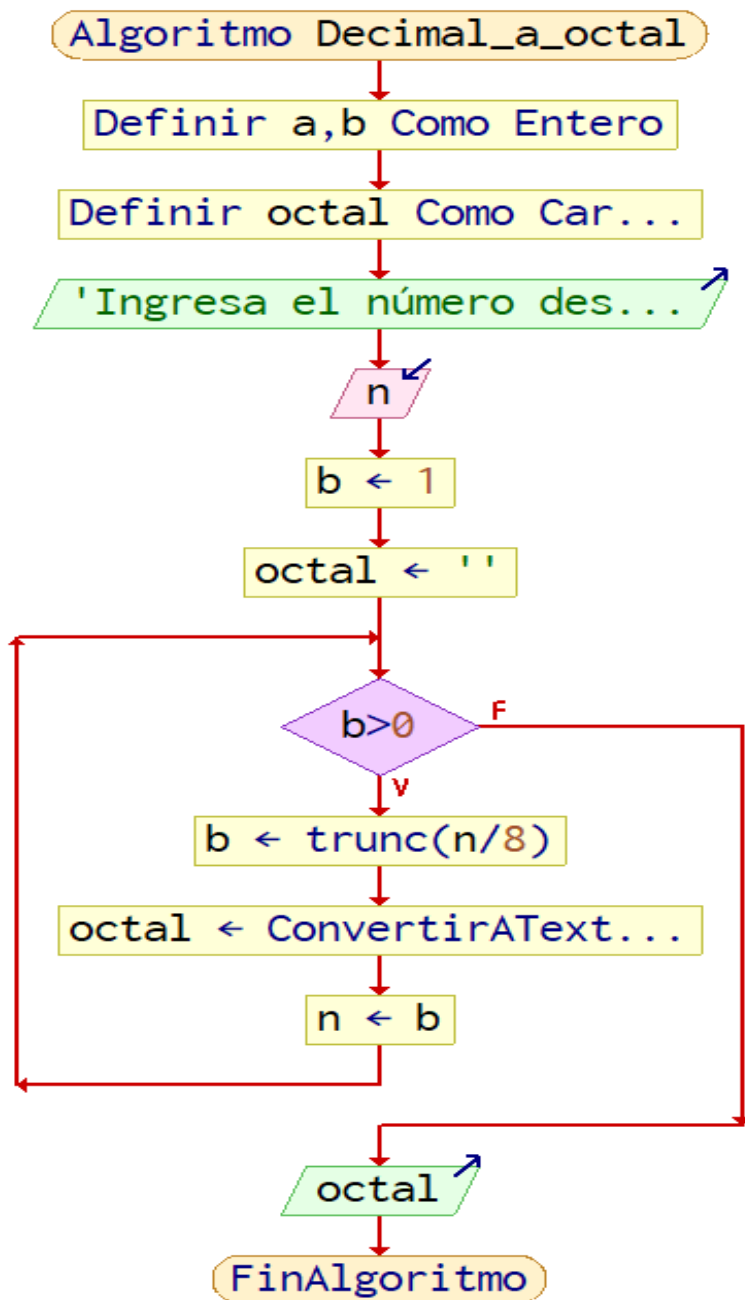
Pruebas de escritorio

Iteración	Decimal		Octal
1	81	81/8=10 residuo 1 10/8=1 residuo 2 1/8=0 residuo 1 121	121
2	676	676/8=84 residuo 4 84/8=10 residuo 4 10/8=1 residuo 2 1/8=0 residuo 1 1244	1244

Pseudocódigo

```
1  Algoritmo Decimal_a_octal
2      definir a,b Como Entero
3      definir octal Como Caracter
4      escribir "Ingresa el número deseado"
5      leer n
6      b= 1
7      octal = ""
8      mientras b > 0 Hacer
9          b = trunc(n/8)
10         octal = ConvertirATexto(n mod 8) + octal
11         n = b
12     FinMientras
13     escribir octal
14
15 FinAlgoritmo
16
```

Diagrama de flujo



Pruebas de escritorio

PSelnt - Ejecutando proceso CONVERSION_DECIMAL_A_OCTAL

```
*** Ejecución Iniciada. ***  
Escriba el número en base decimal  
> 198  
El número en base octal es:306  
*** Ejecución Finalizada. ***
```

Conversiones de octal a binario

Algoritmo

Para convertir de **octal** a **binario** se sigue los siguientes pasos:

1. Solicitamos un número en base 8 y almacenamos la variable.
2. De cada dígito octal, se sustituirá por tres dígitos binarios. La sustitución será la siguiente: el 0 por el 000, el 1 por el 001, el 2 por el 010, el 3 por el 011, el 4 por el 100, el 5 por el 101, el 6 por el 110 y el 7 por el 111.
3. Se escriben los números binarios en el mismo orden que estaban en su base octal
4. Como resultado tendríamos un número binario.
4.1 si el resultado tiene ceros del lado izquierda se eliminan

Pruebas de escritorio

Iteración	Octal		Binario
1	67	6→110 7→111 111110	110111
2	320	3→011 2→010 0→000 11010000	11010000

Pseudocódigo

```
1  SubProceso s ←primerdigito ( c )
2      Definir s Como Caracter
3      Si c="0" Entonces
4          s←""
5      FinSi
6      Si c="1" Entonces
7          s←"1"
8      FinSi
9      Si c="2" Entonces
10         s←"10"
11     FinSi
12     Si c="3" Entonces
13         s←"11"
14     FinSi
15     Si c="4" Entonces
16         s←"100"
17     FinSi
18     Si c="5" Entonces
19         s←"101"
20     FinSi
21     Si c="6" Entonces
22         s←"110"
23     FinSi
24     Si c="7" Entonces
25         s←"111"
26     FinSi
27     Escribir s
28 Fin SubProceso
29
30 SubProceso s ←conver( c )
31     Definir s Como Caracter
32     Si c="0" Entonces
33         s←"000"
34     FinSi
35     Si c="1" Entonces
36         s←"001"
37     FinSi
38     Si c="2" Entonces
39         s←"010"
40     FinSi
41     Si c="3" Entonces
42         s←"011"
43     FinSi
44     Si c="4" Entonces
45         s←"100"
46     FinSi
47     Si c="5" Entonces
48         s←"101"
49     FinSi
50     Si c="6" Entonces
51         s←"110"
52     FinSi
53     Si c="7" Entonces
54         s←"111"
55     FinSi
56     Escribir s
57 FinSubProceso
58
59 Proceso octal_bina
60     Definir oct,t Como Caracter
61     Definir c Como Entero
```

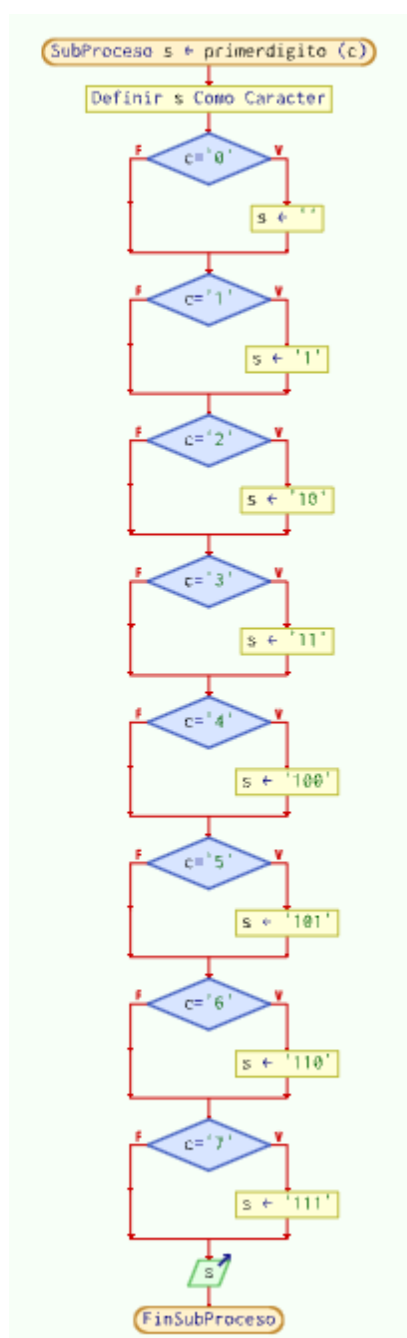


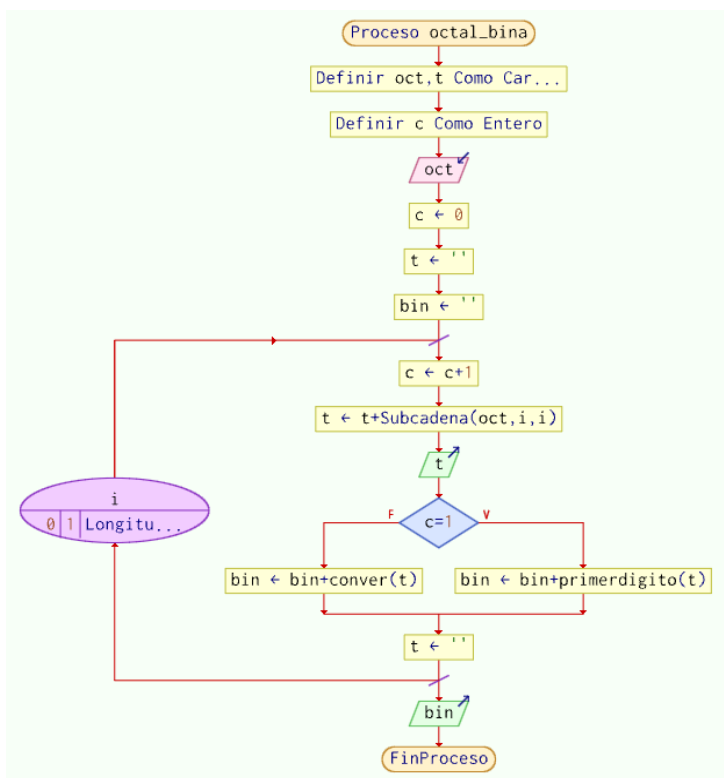
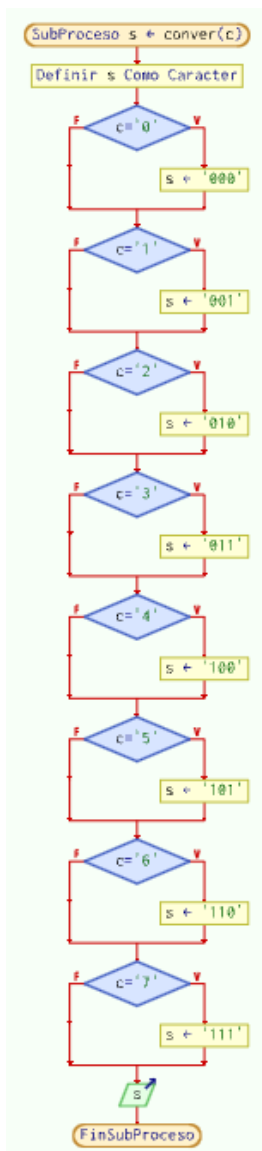
```

62  Leer oct
63  c←0
64  t←""
65  bin←""
66  Para i←0 Hasta Longitud(oct)-1 Con Paso 1 Hacer
67      c←c+1
68      t←t+Subcadena(oct,i,i)
69      Imprimir t
70      Si c=1 Entonces
71          bin←bin+primerdigito(t)
72      SiNo
73          bin←bin+conver(t)
74      FinSi
75      t←""
76  Fin Para
77  Escribir bin
78  FinProceso


```

Diagrama de flujo





Pruebas de escritorio

 PSeInt - Ejecutando proceso CONVERSION_OCTAL_A_BINARIO

*** Ejecución Iniciada. ***

Escriba el número en base octal:

> 367

El número en base binario es:11110111

*** Ejecución Finalizada. ***

Conversiones de binario a octal

Algoritmo

Para convertir de **binario** a **octal** se sigue los siguientes pasos:

- 1) Solicitar un número binario y almacenarlo en una variable.
- 2) Dividir en grupos de tres bits de derecha a izquierda.
- 3) Se agregan los 0 necesarios al inicio del número binario para completar los tres bits.
- 4) Tomar cada grupo de tres bits y realizar la conversión al sistema octal; se sustituirá por tres dígitos binarios. La sustitución será la siguiente: el 0 por el 000, el 1 por el 001, el 2 por el 010, el 3 por el 011, el 4 por el 100, el 5 por el 101, el 6 por el 110 y el 7 por el 111..
- 5) Escribir el nuevo número en el mismo orden en que se realizó la separación.
- 6) Como resultado tendríamos un número en base 8.

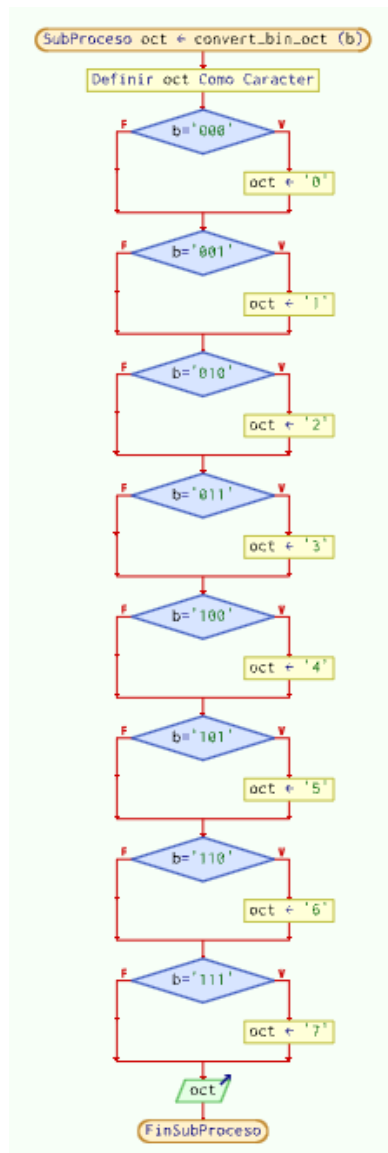
Pruebas de escritorio

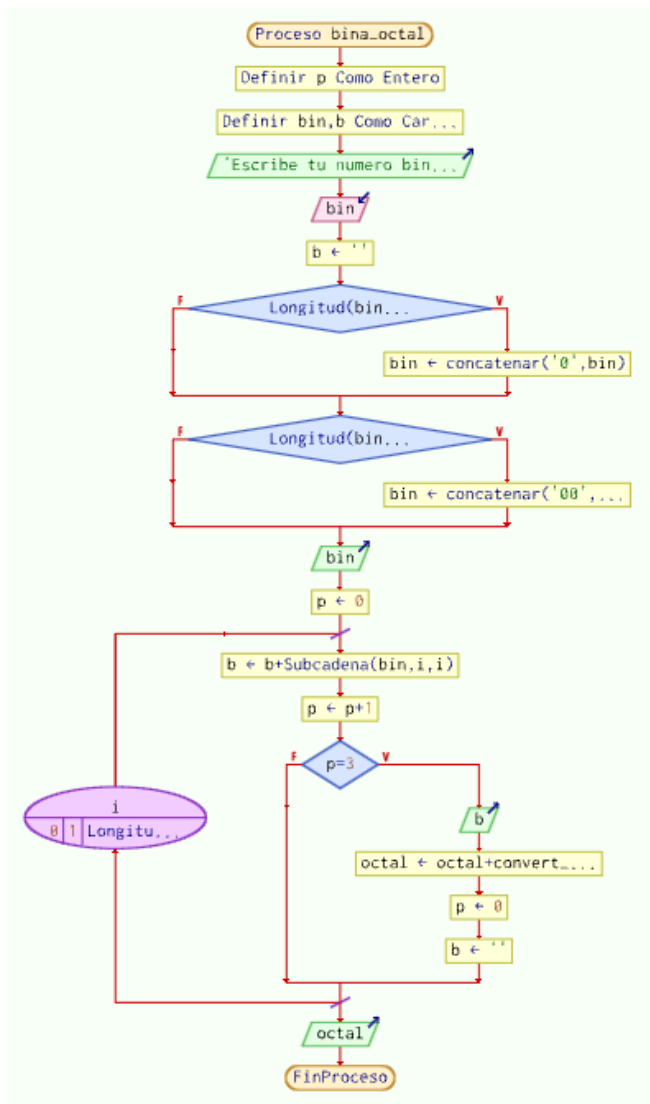
Iteración	Binario		Octal
1	11110	011→3 110→6 36	36
2	001111	0001→1 111→7 17	17

Pseudocódigo

```
1  SubProceso oct ← convert_bin_oct ( b )
2  Definir oct Como Caracter
3  Si b="000" Entonces
4  |   oct←"0"
5  FinSi
6  Si b="001" Entonces
7  |   oct←"1"
8  FinSi
9  Si b="010" Entonces
10 |   oct←"2"
11 FinSi
12 Si b="011" Entonces
13 |   oct←"3"
14 FinSi
15 Si b="100" Entonces
16 |   oct←"4"
17 FinSi
18 Si b="101" Entonces
19 |   oct←"5"
20 FinSi
21 Si b="110" Entonces
22 |   oct←"6"
23 FinSi
24 Si b="111" Entonces
25 |   oct←"7"
26 FinSi
27 Escribir oct
28 Fin SubProceso
29
30 Proceso bina_octal
31 Definir p Como Entero
32
33 Definir bin,b Como Caracter
34 Escribir "Escribe tu numero binario"
35 Leer bin
36 b←""
37 Si Longitud(bin) mod 3=2 Entonces
38 |   bin←concatenar( "0",bin)
39 FinSi
40 Si Longitud(bin) mod 3=1 Entonces
41 |   bin←concatenar("00", bin)
42 FinSi
43 Escribir bin
44 p←0
45 Para i←0 Hasta Longitud(bin)-1 Con Paso 1 Hacer
46 |   b←b+Subcadena(bin,i,i)
47 |   p=p+1
48 |   Si p=3 Entonces
49 |   |   Escribir b
50 |   |   octal←octal+convert_bin_oct(b)
51 |   |   p←0
52 |   |   b←""
53 |   FinSi
54 Fin Para
55 Escribir octal
56 FinProceso
```

Diagrama de flujo





Pruebas de escritorio

```

PSeInt - Ejecutando proceso CONVERSION_BINARIO_A_OCTAL
*** Ejecución Iniciada. ***
Escriba el número en base binario:
> 1110100
El número en base octal es:164
*** Ejecución Finalizada. ***

```

Conversiones de Hexadecimal a binario

Algoritmo

Para convertir de hexadecimal a binario se sigue los siguientes pasos:

- 1) Solicitar un número en base hexadecimal y almacenarlo en una variable
- 2) De cada dígito del valor, se sustituirá por cuatro dígitos binarios. La sustitución será la siguiente: el 0 por el 0000, el 1 por el 0001, el 2 por el 0010, el 3 por el 0011, el 4 por el 0100, el 5 por el 0101, el 6 por el 0110, el 7 por el 0111, el 8 por el 1000 y el 9 por el 1001. y para las letras el A por el

1010, el B por el 1011, el C por el 1100, el D por el 1101, el E por el 1110 y el F por el 1111.

- 3) Se escriben los números binarios en el mismo orden que estaban en su base hexadecimal.
- 4) Como resultado tendríamos un número binario.

Pruebas de escritorio

Iteración	Hexadecimal		Binario
1	26B	2 → 0001 6 → 0110 B → 1011 101101011	101101011
2	921A	9 → 1001 2 → 0010 1 → 0001 A → 1010 1001001000011010	100100100001101 0

Pseudocódigo

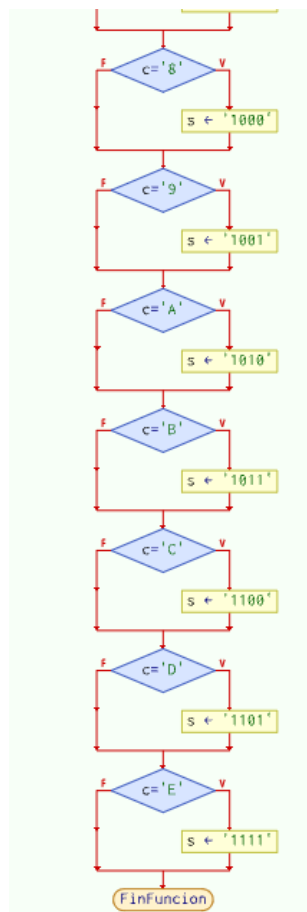
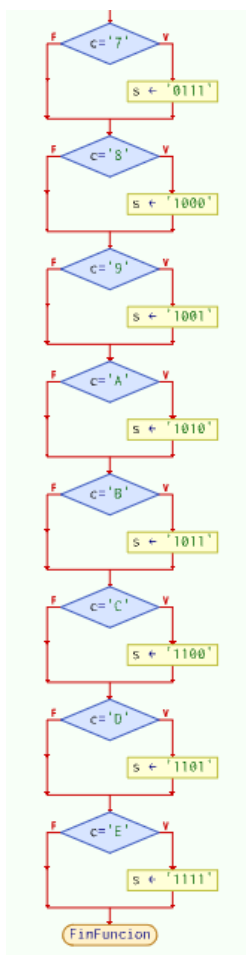
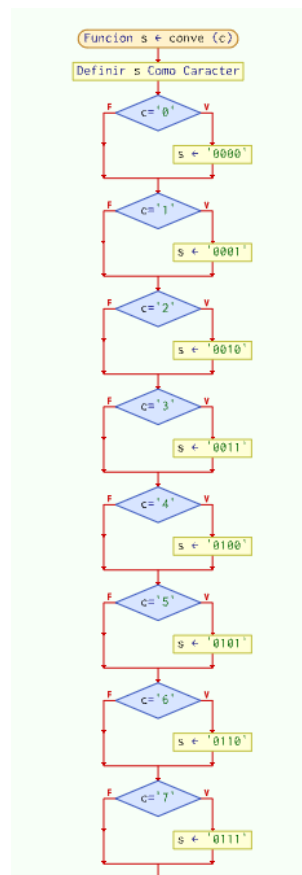
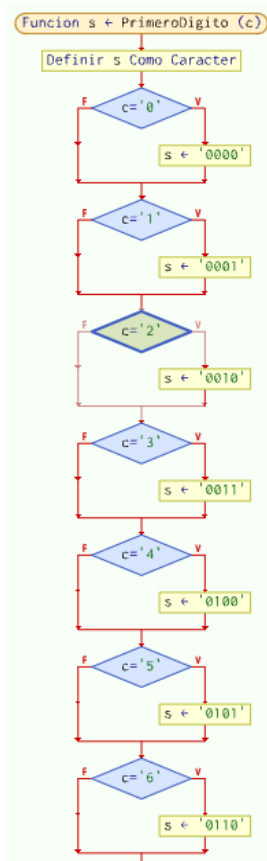
```
1  SubProceso s ← PrimeroDigito (c)
2      Definir s Como Caracter
3      Si c ="0" Entonces
4          s← "0000"
5      FinSi
6      Si c ="1" Entonces
7          s← "0001"
8      FinSi
9      Si c ="2" Entonces
10         s← "0010"
11     FinSi
12     Si c ="3" Entonces
13         s← "0011"
14     FinSi
15     Si c ="4" Entonces
16         s← "0100"
17     FinSi
18     Si c ="5" Entonces
19         s← "0101"
20     FinSi
21     Si c ="6" Entonces
22         s← "0110"
23     FinSi
24     Si c ="7" Entonces
25         s← "0111"
26     FinSi
27     Si c ="8" Entonces
28         s← "1000"
29     FinSi
30     Si c ="9" Entonces
31         s← "1001"
32     FinSi
33     Si c ="A" Entonces
34         s← "1010"
35     FinSi
36     Si c ="B" Entonces
37         s← "1011"
38     FinSi
39     Si c ="C" Entonces
40         s← "1100"
41     FinSi
42     Si c ="D" Entonces
43         s← "1101"
44     FinSi
45     Si c ="E" Entonces
46         s← "1111"
47     FinSi
48 FinSubProceso
49 SubProceso s ← conve (c)
50     Definir s Como Caracter
51     Si c ="0" Entonces
52         s← "0000"
53     FinSi
54     Si c ="1" Entonces
55         s← "0001"
56     FinSi
57     Si c ="2" Entonces
58         s← "0010"
```

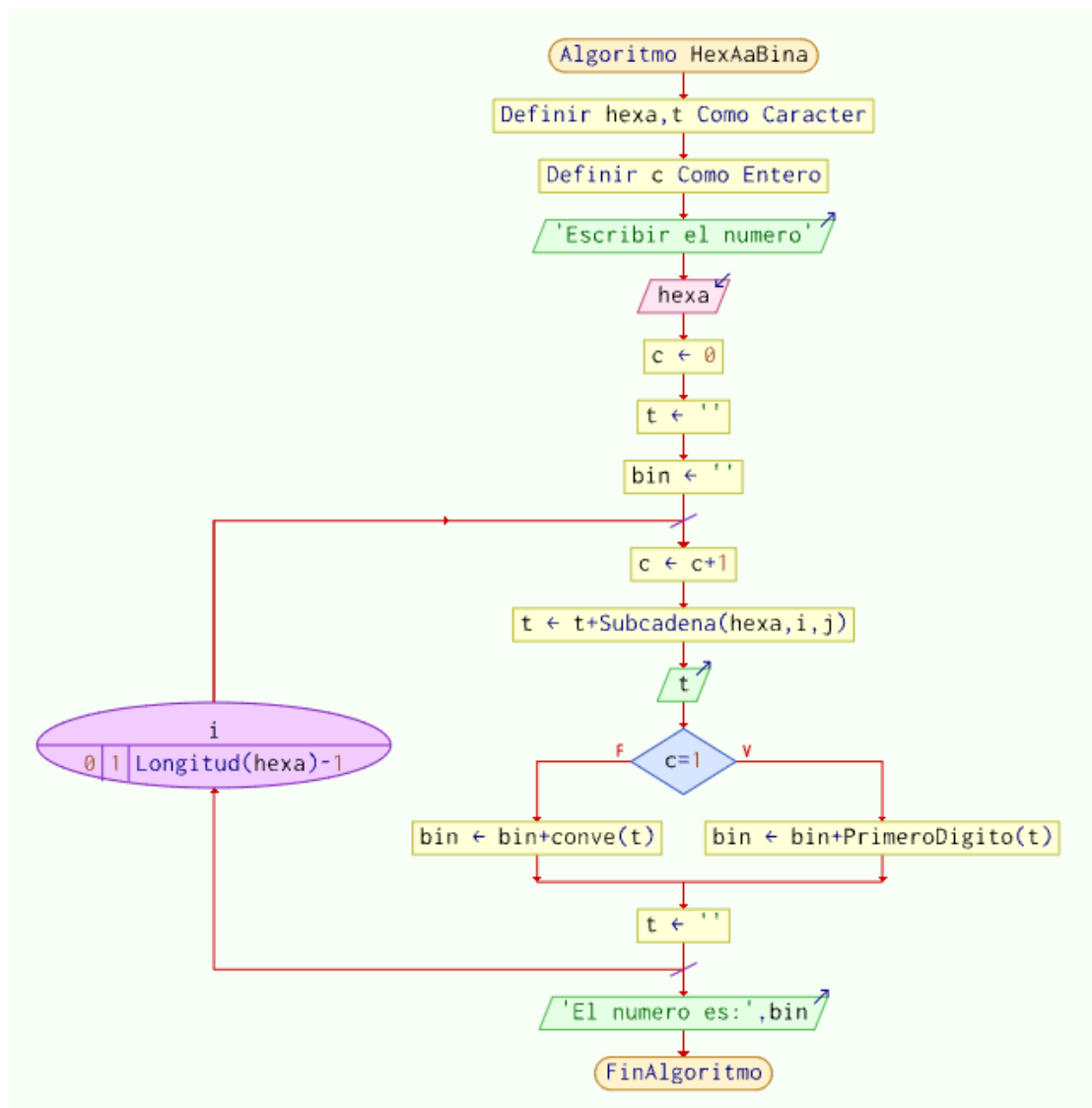
```

59     FinSi
60     Si c ="3" Entonces
61         s← "0011"
62     FinSi
63     Si c ="4" Entonces
64         s← "0100"
65     FinSi
66     Si c ="5" Entonces
67         s← "0101"
68     FinSi
69     Si c ="6" Entonces
70         s← "0110"
71     FinSi
72     Si c ="7" Entonces
73         s← "0111"
74     FinSi
75     Si c ="8" Entonces
76         s← "1000"
77     FinSi
78     Si c ="9" Entonces
79         s← "1001"
80     FinSi
81     Si c ="A" Entonces
82         s← "1010"
83     FinSi
84     Si c ="B" Entonces
85         s← "1011"
86     FinSi
87     Si c ="C" Entonces
88         s← "1100"
89     FinSi
90     Si c ="D" Entonces
91         s← "1101"
92     FinSi
93     Si c ="E" Entonces
94         s← "1111"
95     FinSi
96 FinSubProceso
97 PROCESO HexAaBina
98     Definir hexa, t Como Caracter
99     Definir c Como Entero
100     Escribir "Escribir el numero"
101     Leer hexa
102     c←0
103     t←""
104     bin←""
105     Para i <- 0 Hasta Longitud(hexa)-1 Con Paso 1 Hacer
106         c ← c+1
107         t ← t+Subcadena(hexa,i,j)
108         Imprimir t
109
110         Si c = 1 Entonces
111             bin ←bin + PrimeroDigito(t)
112         SiNo
113             bin← bin+conve(t)
114         FinSi
115         t←""
116     Fin Para
117     Escribir "El numero es:" ,bin;
118 FinProceso

```

Diagrama de flujo





Pruebas de escritorio

```

Escriba el número en base Hexadecimal :
PSeInt - Ejecutando proceso CONVERSION_HEXADECIMAL_A_BINARIO
*** Ejecución Iniciada. ***
Escriba el número en base Hexadecimal :
> D4B
El número en Binario es:110101001011
*** Ejecución Finalizada. ***
  
```

Conversiones de binario a hexadecimal

Algoritmo

Para convertir de **binario** a **hexadecimal** se sigue los siguientes pasos:

- 1) Solicitar un número binario y almacenarlo en una variable.
- 2) Dividir en grupos de cuatro bits de derecha a izquierda.
- 3) Tomar cada grupo de cuatro bits y realizar la conversión al sistema decimal: el 0 por el 0000, el 1 por el 0001, el 2 por el 0010, el 3 por el 0011, el 4 por el 0100, el 5 por el 0101, el 6 por el 0110, el 7 por el 0111, el 8 por el 1000 y el 9 por el 1001. y para las letras el A por el 1010, el B por el 1011, el C por el 1100, el D por el 1101, el E por el 1110 y el F por el 1111.
- 4) Escribir el nuevo número en el mismo orden en que se realizó la separación.
- 5) Como resultado tendríamos un número en base 16.

Pruebas de escritorio

Iteración	Binario		Hexadecimal
1	1110010101	0011 1001 0101 0011 →3 1001 →9 0101 →5 395	395
2	110101101	0001 1010 1101 0001 →1 1010 →A 1101 →D 1AD	1AD

Pseudocódigo

```
1  Funcion s ← PrimeroDigito (c)
2  Definir s Como Caracter
3  Si c="0000" Entonces
4  ..... s ← "0"
5  FinSi
6  Si c="0001" Entonces
7  ..... s ← "1"
8  FinSi
9  Si c="0010" Entonces
10 ..... s ← "2"
11 FinSi
12 Si c="0011" Entonces
13 ..... s ← "3"
14 FinSi
15 Si c="0100" Entonces
16 ..... s ← "4"
17 FinSi
18 Si c="0101" Entonces
19 ..... s ← "5"
20 FinSi
21 Si c="0110" Entonces
22 ..... s ← "6"
23 FinSi
24 Si c="0111" Entonces
25 ..... s ← "7"
26 FinSi
27 Si c="1000" Entonces
28 ..... s ← "8"
29 FinSi
30 Si c="1001" Entonces
31 ..... s ← "9"
32 FinSi
33 Si c="1010" Entonces
34 ..... s ← "A"
35 FinSi
36 Si c="1011" Entonces
37 ..... s ← "B"
38 FinSi
39 Si c="1100" Entonces
40 ..... s ← "C"
41 FinSi
42 Si c="1101" Entonces
43 ..... s ← "D"
44 FinSi
45 Si c="1111" Entonces
46 ..... s ← "E"
47 FinSi
48 FinFuncion
49 Funcion s ← Conve (c)
50 Definir s Como Caracter
51 Si c="0000" Entonces
52 ..... s ← "0"
53 FinSi
54 Si c="0001" Entonces
55 ..... s ← "1"
56 FinSi
57 Si c="0010" Entonces
58 ..... s ← "2"
59 FinSi
60 Si c="0011" Entonces
61 ..... s ← "3"
62 FinSi
63 Si c="0100" Entonces
64 ..... s ← "4"
65 FinSi
66 Si c="0101" Entonces
67 ..... s ← "5"
68 FinSi
```

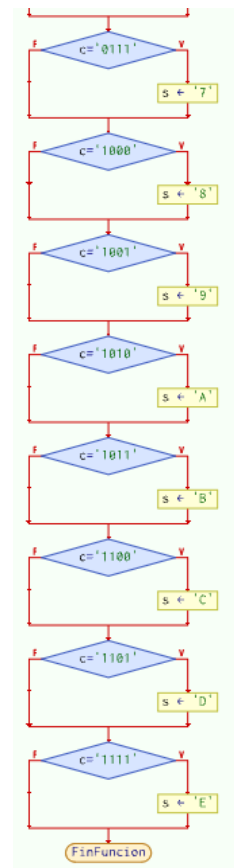
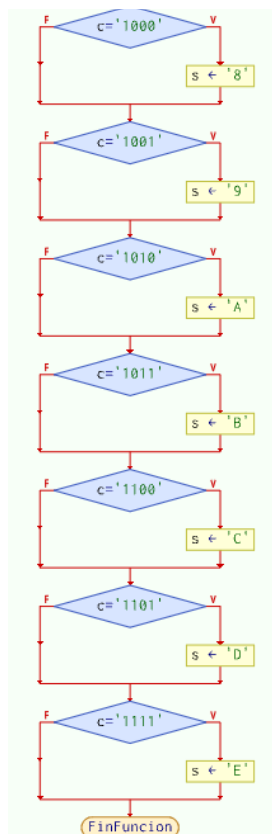
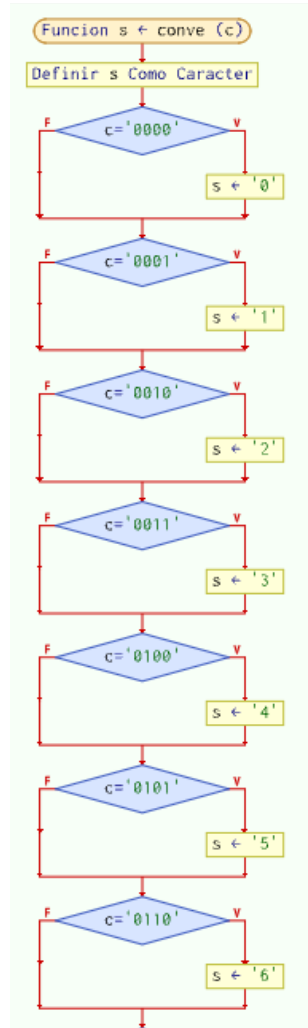
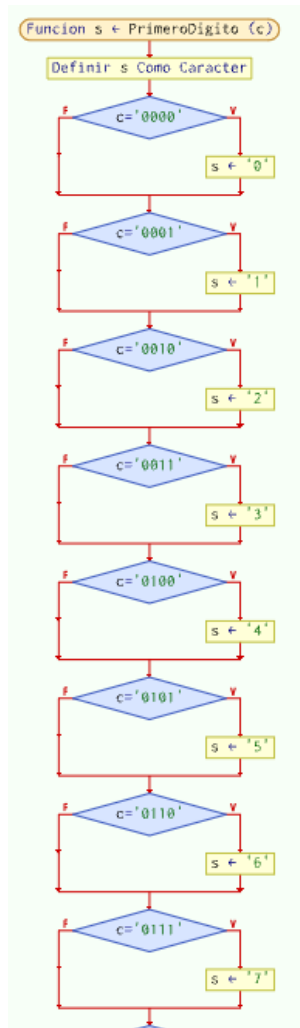
```

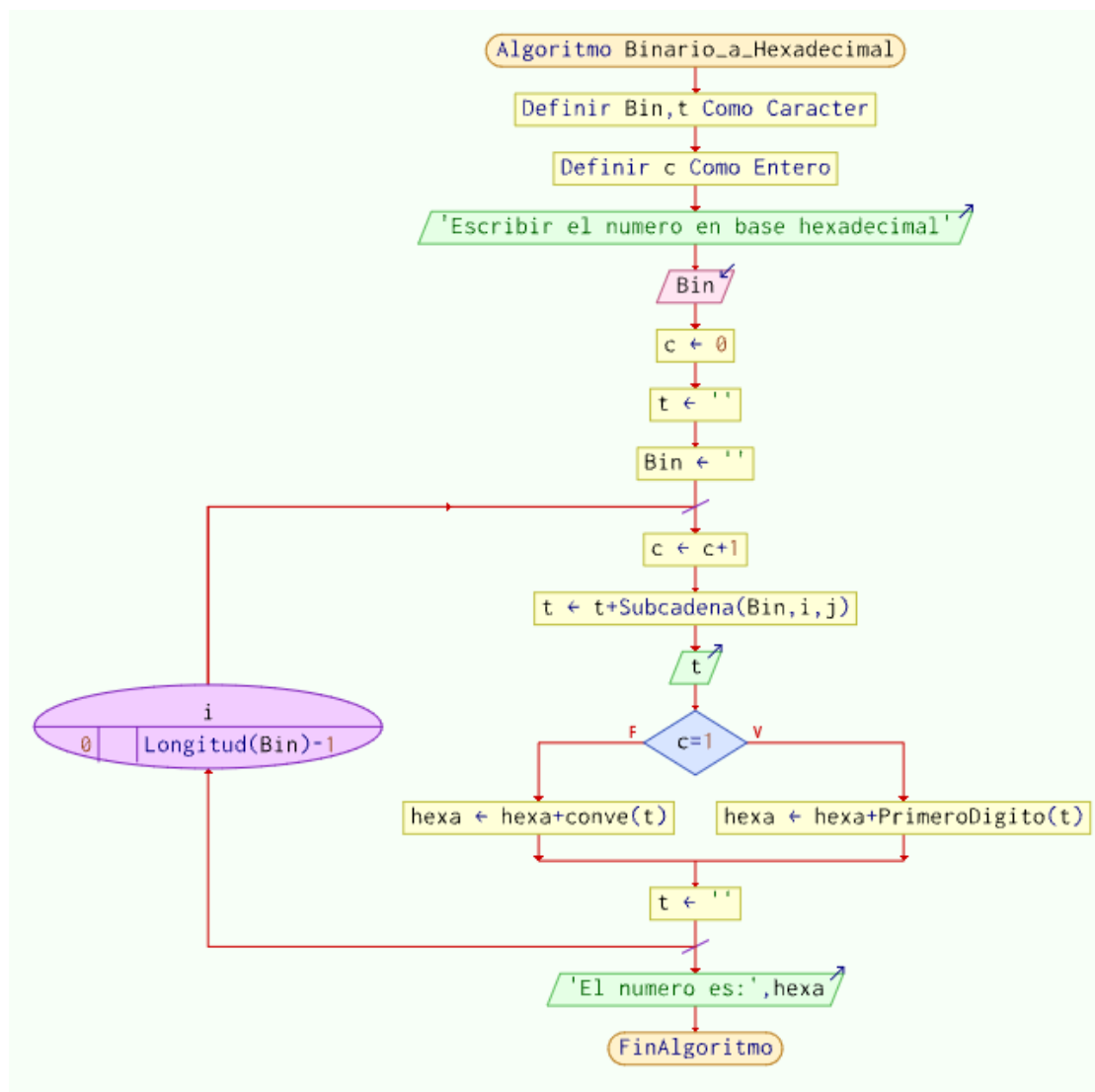
69     Si c="0110" Entonces
70         s ← "6"
71     FinSi
72     Si c="0111" Entonces
73         s ← "7"
74     FinSi
75     Si c= "1000" Entonces
76         s ← "8"
77     FinSi
78     Si c= "1001" Entonces
79         s ← "9"
80     FinSi
81     Si c= "1010" Entonces
82         s ← "A"
83     FinSi
84     Si c= "1011" Entonces
85         s ← "B"
86     FinSi
87     Si c="1100" Entonces
88         s ← "C"
89     FinSi
90     Si c="1101" Entonces
91         s ← "D"
92     FinSi
93     Si c="1111" Entonces
94         s ← "E"
95     FinSi
96 FinFuncion

97
98 Algoritmo HexAaBina
99     Definir hexa,t Como Caracter
100     Definir c Como Entero
101     Escribir 'Escribir el numero'
102     Leer hexa
103     c ← 0
104     t ← ''
105     bin ← ''
106     Para i←0 Hasta Longitud(hexa)-1 Hacer
107         c ← c+1
108         t ← t+Subcadena(hexa,i,j)
109         Escribir t
110         Si c=1 Entonces
111             bin ← bin+PrimerDigito(t)
112         SiNo
113             bin ← bin+conve(t)
114         FinSi
115         t ← ''
116     FinPara
117     Escribir 'El numero es:',bin
118 FinAlgoritmo
119

```

Diagrama de flujo





Pruebas de escritorio

PSelnt - Ejecutando proceso CONVERSION_BINARIO_A_HEXADECIMAL

```

*** Ejecución Iniciada. ***
Escriba el número en base Binaria :
> 110110111
El número en base Hexadecimal es:1B7
*** Ejecución Finalizada. ***

```

Conclusiones:

Conclusión grupal: En este proyecto se utilizó todo lo aprendido a lo largo del semestre, como por ejemplo: los ciclos de repetición for, while y do while; las estructuras if, else-if y switch para realizar diferentes conversiones a diferentes valores numéricos como el binario, decimal, octal y hexadecimal.

Las funciones se fueron construyendo desde el algoritmo, las pruebas de escritorio y el pseudocódigo para terminar creando los archivos .c, .h y main.c en lenguaje .C creando las funciones respectivas a los diferentes tipos de variables.

Conclusiones individuales:

Arredondo Granados Gerardo: El proyecto 1 constaba de funciones las cuales puedan hacer conversiones entre varios tipos de números, realizar los pseudocódigos y las descripciones de estas funciones, se puede decir que los objetivos se han cumplido, para llegar a completar estas funciones, se hizo uso de los temas vistos a inicio de semestre, cosas como separar un archivo en varias funciones, y en sí separar el código en varios archivos, se hizo uso de un archivo .h y .c.

Con esto concluye el proyecto 1, conversiones de tipos de números.

Casillas Herrera Leonardo Didier: Con este proyecto pusimos en práctica todo lo que aprendimos desde los algoritmos, las pruebas de escritorio, los pseudocódigos y diagramas de flujo para finalmente poder hacer el código en C de las diferentes conversiones que necesitábamos, utilizamos varias funciones con diferentes variables para realizar cada conversión. Finalmente todo funcionó correctamente por lo que este proyecto fue de mucha ayuda para ejemplificar el uso de todo lo que se nos enseñó anteriormente.

Diaz Gonzalez Rivas Angel Iñaqui: Al realizar los programas se utilizaban los ciclos for, while, así como if, else if para poder obtener el número esperado. Switch sirvió al momento de que el usuario ingresara cualquier dato el programa a partir de los case podía proporcionar una respuesta adecuada. También se llegaron a utilizar muchos tipos de valores ya sea int, char, void, etc. para funciones que requería el programa, como por ejemplo, los valores en hexadecimal no podían expresarse como enteros, por lo que era necesario utilizar caracteres.

En general se podría decir que el proyecto terminó en un programa que podía manejar las conversiones de valores en decimal, binario, octal y hexadecimal.

Galván Castro Martha Selene: En conclusión en este proyecto se pudo solucionar el problema planteado debido a que se logró realizar de manera exitosa el proceso de crear conversor de bases numéricas, se obtuvo una ejecución correcta en todas las conversiones.

Al final se logró crear el programa en C de manera correcta, utilizando los distintos bucles y las conversiones bases numéricas utilizada: hexadecimal, decimal, octal y binario estuvieron bien realizadas.