



UNAH

UNIVERSIDAD NACIONAL
AUTÓNOMA DE HONDURAS

IS-913, Diseño de Compiladores

Tercer Periodo 2021

Informe del Proyecto de Clase

Elaborado por:

20151021802	Abner Eliezer Betancourt
20161001202	Andrés Fernando Lizardo Rodríguez
20151001680	Fidel Ernesto García Gutiérrez
20141031775	Jose Carlos Velásquez
20161001463	Roger Alfredo Molina
20161003401	Víctor Elías Vásquez Padilla
20131000142	Tomasa Clementina Meza

Catedrático: Alex Hernán Moncada

Contenido

Contenido	2
Introducción	2
Definición del Proyecto: Proyecto Convertidor de Números	9
Analizador Léxico	10
Analizador Sintáctico	11
Capturas del programa en ejecución	12
Hoja de Desempeño	13
Bibliografía	14

Introducción

Un sistema de numeración es el conjunto de símbolos y reglas que se utilizan para la representación de datos numéricos o cantidades. Un sistema de numeración se caracteriza por su base, que es el número de símbolos distintos que utiliza y además es el coeficiente que determina cuál es el valor de cada símbolo dependiendo de la posición que ocupe.

Dentro de este sistema existen dos clasificaciones que permiten catalogar algunas de las características más relevantes, las cuales son:

- **No posicionales:** El valor de los símbolos que componen al sistema es fijo y no dependen de la posición que ocupa el símbolo dentro del número, un ejemplo de ello son los números romanos.
- **Posicionales:** El valor de los símbolos que componen el sistema, depende del valor que se les ha asignado y de la posición que ocupan en el número, un ejemplo de ello son los números decimales.

En el presente informe se presentan los siguientes sistemas binarios.

Sistema Binario

Es un sistema posicional que utiliza sólo dos símbolos para representar números llamados BITS, el cual puede tomar los valores de 1(encendido) o 0(cero, apagado).

Ventajas:

- Al poseer solo dos símbolos o estados, es la ventaja fundamental que aplica al manejo de información de computadores, pues es más sencillo que un computador entienda dos estados de voltaje (cada uno representaría un 1 o un 0)
- Gracias al empleo de técnicas de codificación, la transmisión de información en sistema binario es más confiable, ya que permite la detección de fallas y si el código lo permite la corrección de la misma.
- Gracias a la aplicación de conversores entre sistemas digitales otros sistemas de numeración y sistemas analógicos (ADC y DAC), es posible la transmisión de información analógica (voz, audio y video) y texto (código ASCII, BCD, entre otros) a través de información binaria, y por consiguiente el manejo de todo tipo de información a través en un computador digital.

Desventajas:

- En comparación con otros sistemas de numeración, al poseer solo 2 símbolos, para la representación de alguna cifra son requeridas mayor cantidad de dígitos binarios que algún otro sistema cuya base sea mayor que 2.
- Al no poseer números negativos, es necesario el uso de operaciones matemáticas como son el complemento al uno y complemento al dos, para así poder realizar operaciones básicas de sustracción.
- Para sistemas que requieran alta precisión numérica, el manejo de números no enteros, números muy grandes o muy pequeños; se necesita de operaciones más elaboradas como son el uso de coma flotante (norma ANSI/IEEE-754) para la equivalencia entre dicho número en notación científica a sistema de numeración binario.

Sistema Decimal

Su nombre hace referencia a que siempre se forman grupos de 10:

- Las unidades son elementos sin agrupar.
- Las decenas son grupos de diez unidades.
- Las centenas son grupos de diez decenas.
- Las unidades de mil son grupos de diez centenas.
- Las decenas de mil son 10 grupos de unidades de mil.
- Las centenas de mil son grupos de 10 decenas de mil.

Ventajas

- Uso actual mundial en toda área que requiera manejo de información numérica debido a su fácil entendimiento y manejo de reglas.
- Al ser un sistema de numeración posicional, reemplazo en su mayoría a los otros sistemas de numeración no posicionales, como el sistema de numeración romano, el cual es no posicional y no posee el número 0 o elemento neutro, y al resto de los sistemas de numeración posicionales pero cuya simbología, reglas y bases requerían mayor tratamiento (sistemas maya, egipcio, griego y babilónico).
- Aplicación en todo tipo de ciencias e ingeniería, y de tratamiento más profundo en cuanto a las posibles estructuras que se pueden optar para un manejo óptimo en el área de aplicación (número real, entero, fraccionario, racional,

irracional, exacto, periódico, número complejo, notación científica)

Desventajas

En el área de la informática y computación, los cálculos y procedimientos no son posibles realizarlos en base 10, es necesario el uso de sistema binario (base 2), el cual solo maneja 2 posibles símbolos (0 o 1). No es posible la elaboración de un sistema computacional en sistema decimal, ya que esto requeriría que un sistema lógico diferenciara entre 10 niveles de voltaje para el entendimiento de los procedimientos, para luego elaborar lógica circuito con dichos niveles (ineficiente).

A pesar de su amplia aceptación, para usos más específicos como la medición de tiempos o geo-localizaciones, son empleados sistemas de numeración como el sexagesimal (grados, minutos y segundos), aunque son empleadas técnicas de conversiones equivalentes pero más tediosas para el entendimiento entre dichos sistemas.

Debido a que su base (10) no es múltiplo de las bases de los sistemas binario (base 2), octal (base 8) y hexadecimal (base 16), no existe un método directo o de sustitución para realizar conversiones entre el sistema decimal y los mencionados (siempre es necesario procedimientos aritméticos para la conversión).

Sistema Octal

En el sistema octal, los números se representan mediante ocho dígitos diferentes: 0, 1, 2, 3, 4, 5, 6 y 7. Cada dígito tiene, naturalmente, un valor distinto dependiendo del lugar que ocupen. El valor de cada una de las posiciones viene determinado por las potencias de base 8. La conversión de un número decimal a octal, y viceversa, se realiza del mismo modo que la de los números binarios, aunque, lógicamente, se emplea como base el número 8 en vez del 2.

Ventajas:

- El sistema de numeración octal es muy útil al representar números de bits múltiples, debido a que su base es potencia de 2. Puesto que cada cadena de 3 bits puede tomarse en ocho diferentes combinaciones, cada cadena de 3 bits puede representarse de manera única mediante un dígito octal. De este modo

es muy fácil de convertir un número binario a octal separando los bits en cadenas de tres y reemplazando cada grupo con el correspondiente dígito octal.

- La numeración octal es tan buena como la binaria y la hexadecimal para operar con fracciones, puesto que el único factor primo para sus bases es 2. Todas las fracciones que tengan un denominador distinto de una potencia de 2 tendrán un desarrollo octal periódico.
- Tiene la ventaja de que no requiere utilizar otros símbolos diferentes de los dígitos.

Desventajas:

- El desuso del sistema octal se debe a la preponderancia de las máquinas que procesan bytes compuestos de 8 bits.
- Es difícil extraer los valores de byte individual en cantidades de bytes múltiples en la representación octal.
- En comparación con el sistema hexadecimal, no agrupa la misma cantidad de información debido a que posee una base mucho menor si se compara con una cifra Hex con la misma cantidad de dígitos.

Sistema Hexadecimal

En este sistema, los números se representan con dieciséis símbolos: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E y F. Se utilizan los caracteres A, B, C, D, E y F representando las cantidades decimales 10, 11, 12, 13, 14 y 15 respectivamente, porque no hay dígitos mayores que 9 en el sistema decimal. El valor de cada uno de estos símbolos depende, como es lógico, de su posición, que se calcula mediante potencias de base 16.

Ventajas:

- Debido a que su base de numeración (base 16), es múltiplo de la base del sistema binario (base 2), la conversión entre dichos sistemas se puede llevar a cabo mediante la técnica de sustitución directa; lo cual representa una ventaja significativa al momento de trabajar entre ambos sistemas.

- El sistema hexadecimal es un sistema de numeración vinculado a la informática, se debe a que un día y tu hexadecimal representa cuatro dígitos binarios exactamente, o un denominado nibble. Los ordenadores interpretan los lenguajes de programación en bytes, y cómo va ahí se compone de ocho dígitos binarios o dos nibbles, es este sistema desplazó al sistema octal para el funcionamiento interno de los computadores actuales

Desventajas:

- Al poseer un número de símbolos mayor al 10, se requiere el uso de símbolos alfanuméricos, por lo que las labores de cálculo y aritmética en este sistema son más tediosas en comparación con los sistemas binarios, octal y decimal.

Sistema Romano

La numeración romana es el sistema que usaban los romanos, actualmente se emplea en fechas, capítulos, etc. El sistema de numeración romana expresa los números por medio de siete letras del alfabeto latino, que son:

I = 1, V = 5, X = 10, L = 50, C = 100, D = 500 y M = 1.000.

Estos valores quedan multiplicados por mil al superponer una raya sobre la correspondiente letra, y por un millón, si se colocan dos rayas.

M = 1 000 000

Ninguna cifra puede repetirse más de tres veces seguidas.

III = 3 XXX = 30 CCC = 300

Las letras V, L y D no pueden duplicarse, porque el doble de éstas son: X, C y M.

Si se coloca una cifra a la derecha de otra siendo su valor menor o igual que ésta sus valores se suman.

VII = 7 XX = 20 CLXVIII = 168

Si se coloca una cifra menor a la izquierda de otra, los valores de ambas se restan.

$$\text{IV} = 4 \quad \text{IX} = 9 \quad \text{XL} = 40$$

Ventajas

- Cualquier número menor a 1000 puede escribirse con 3 símbolos.
- Se utilizan sólo siete símbolos (letras).
- No es posicional; los valores de los símbolos no dependen de su posición.

Desventajas

- No existe el número 0.
- No tiene números negativos.
- No hay fracciones en la este tipo de numeración.

Sistema Quinario

El sistema quinario o también llamado pentario, es un sistema de numeración de base 5 constante, utilizando como símbolos los dígitos 0, 1, 2, 3 y 4.

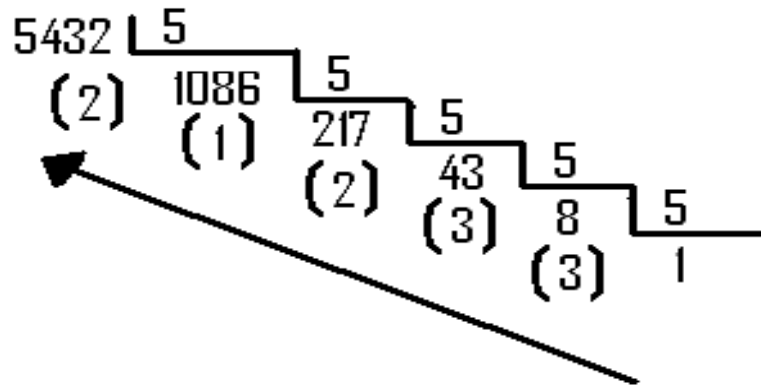
Este sistema tiene su origen en el hecho de que los humanos tienen cinco dedos en cada mano, por lo que es uno de los sistemas de numeración más antiguos. Se cree que los Náhuatl utilizaban este sistema también, o al menos una dicción similar, teniendo en cuenta que su numeración era de manera:

- ce (1)
- ome (2)
- ěyi (3)
- nawi (4)
- macuilli (5)
- chicu-ace (**6 = 5+1**)

Hasta el siglo XX, existían varias tribus en África como los grupos étnicos Bantu y entre estos, los Chewa o los pueblos Gbaya, que todavía seguían utilizando este sistema, sin embargo, el sistema decimal fue tomando más relevancia por medio de la comunicación entre las culturas que ya se habían adaptado a este sistema y eventualmente estos también se adaptaron.

Para convertir un número decimal al sistema quinario, se toma el número y se divide

entre 5 sucesivamente, tomando en cuenta el último cociente y todos los residuos como cifras del número resultante en base 5. Por ejemplo, si quisiéramos convertir 5432 de números decimales a quinarios, se realizaría de la siguiente manera:



Para convertir, se toma en cuenta desde el último cociente, el cuál sería 1, y se cuenta hasta arriba con residuos enmarcados en la imagen.

Por lo tanto, el número resultante sería **133212 en base quinal**.

Definición del Proyecto: Proyecto Convertidor de Números

Cadenas de números enteros con base decimal (base 10) seguidas de un identificador de conversión de destino los cuales pueden ser:

- Hexadecimal.
- Octal.
- Binario.
- Romano.
- Alternativo (uno inventado o agregado por el grupo).
- Aleatorio (toma uno de los destinos de forma aleatoria).

Analizador Léxico

Definición

El analizador léxico es la primera fase de un compilador.

Su principal función consiste en leer los caracteres de entrada y elaborar como salida una secuencia de componentes léxicos que utiliza el analizador sintáctico para hacer el análisis. Esta interacción, suele aplicarse convirtiendo al analizador léxico en una subrutina o corrutina del analizador sintáctico. Recibida la orden "obtén el siguiente componente léxico" del analizador sintáctico, el analizador léxico lee los caracteres de entrada hasta que pueda identificar el siguiente componente léxico.

Los patrones utilizados para la identificación de lexemas:

Patrones:

NUMERO = r'[0-9]+'

DECIMAL = r'Dec'

OCTAL = r'Oct'

HEXADECIMAL = r'Hex'

ROMANO = r'Rom'

BINARIO = r'Bin'

QUINARIO = r'Quint'

Tabla de Tokens, Lexemas y Patrones

Token	Patrón	Ejemplo
id	(digito+sistema\s.)*	20Hex 30Rom
numero	[0-9]	1
numeros	numero+	2030
sistema	(Hex Oct Quint Rom Dec Bin){1}	Hex

Ejemplo de Análisis Sintáctico

Valor	Columna	Tipo Token
144Hex	0	Id
156Bin	7	id
43Oct	14	id

Analizador Sintáctico

Definición

La tarea del analizador es, en este caso, la descomposición y transformación de las entradas en un formato utilizable para su posterior procesamiento. Se analiza una cadena de instrucciones en un lenguaje de programación y luego se descompone en sus componentes individuales.

Reglas de Producción

LL1: " es conjunto vacío

E --> TE'

E'--> "

T --> FT'

T'--> "

F --> id

Tabla de Análisis Sintáctico (LL1)

Primero	Siguiente	No terminal	id	\$
{id}	{\$}	E	E -> T E'	
{",id}	{\$}	E'	E' -> TE'	E' -> "
{id}	{\$,id}	T	T -> F T'	
{'"}	{\$,id}	T'	T' -> "	T' -> "
{id}	{\$,id}	F	F -> id'	

Validación de la Tabla de Análisis Sintáctico LL1

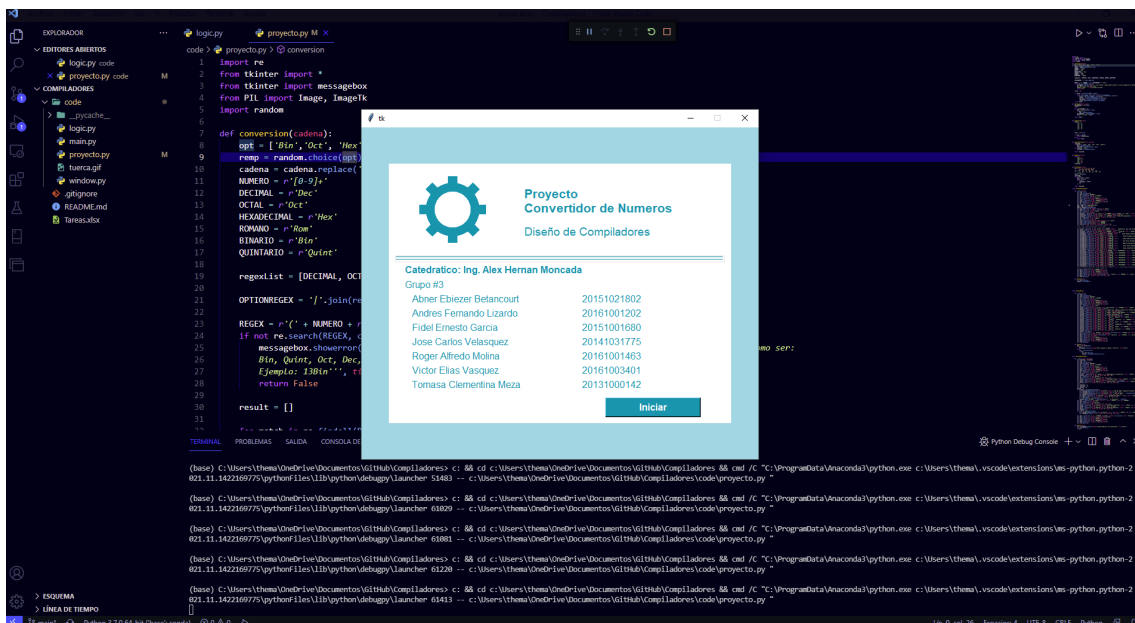
Pila	Entrada	Dominio
\$ E	id \$	
\$ E' T	id \$	E -> T E'
\$ E' T' F	id \$	T -> F T'
\$ E' T' id	id \$	F -> id
\$ E' T'	\$	
\$ E'	\$	T' -> "
\$	\$	E' -> "

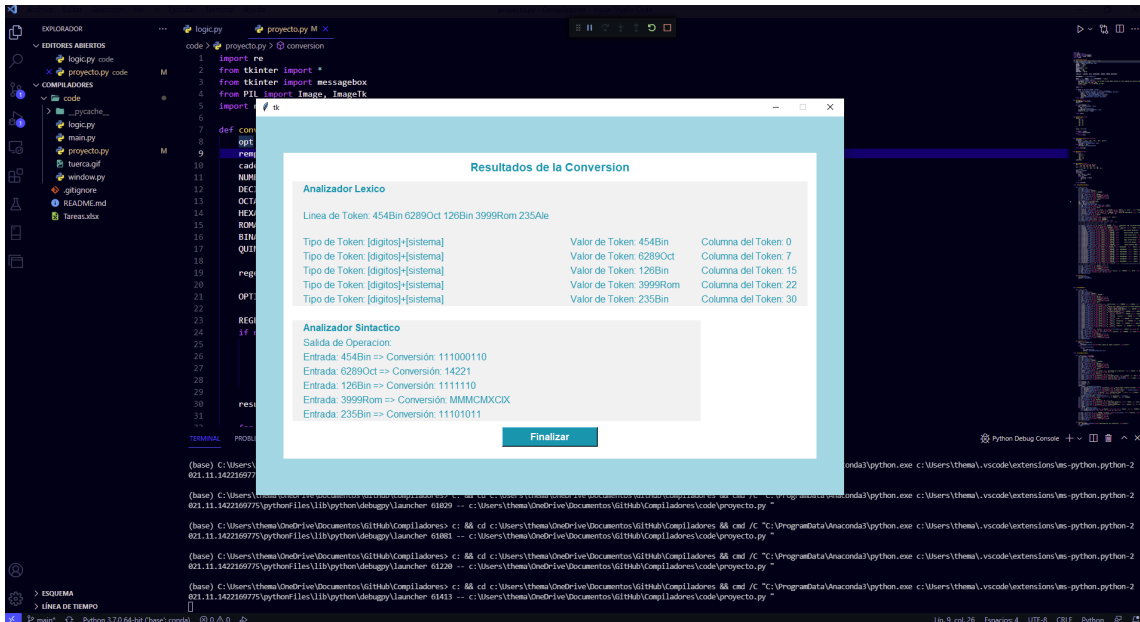
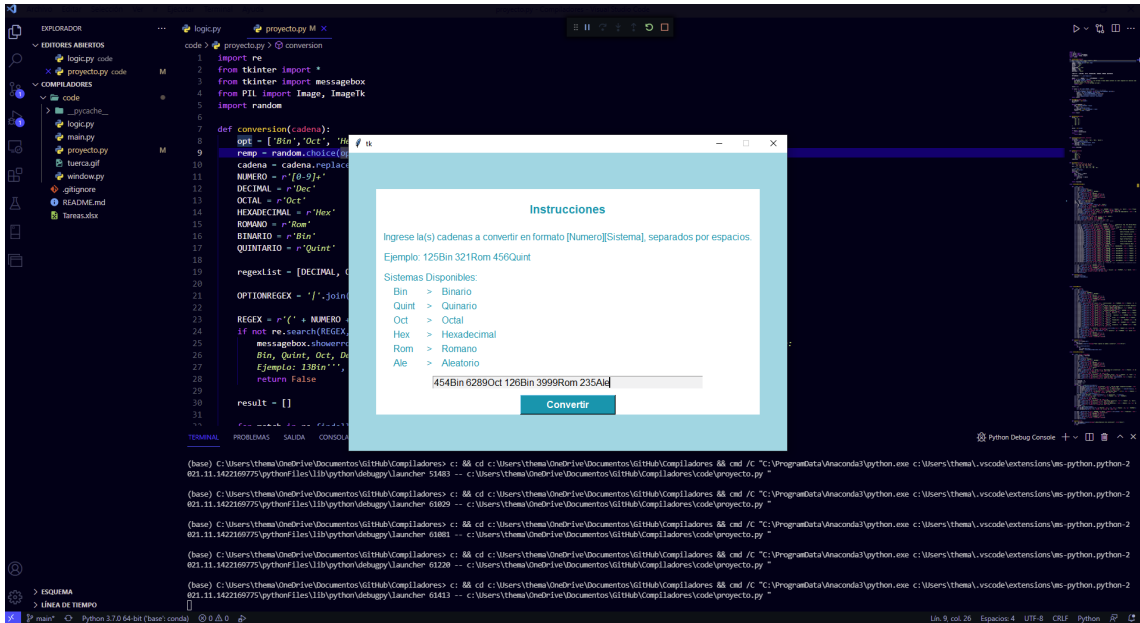
Nombres de las funciones utilizadas:

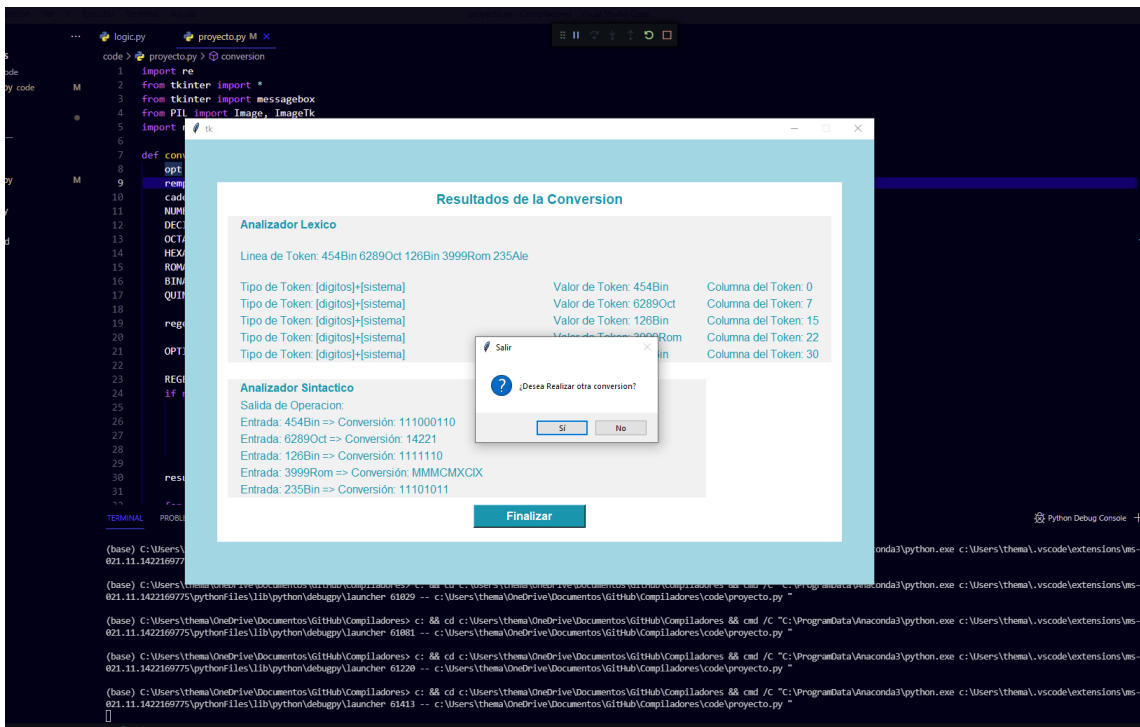
- **def convert(cadena, winConvert):** convierte la cadena de entrada de texto (asumo que la parte en los lexemas y tokens).
- **def decToBase(number, base):** convierte un número de base decimal a un número de base n.

- **def getHexValue(value):** obtener el valor hexadecimal del valor de entrada ingresado.
- **def checkConversion(numero, destino):** revisión de la conversión del valor numero.
- **def getBase(destino):** Obtiene la base del valor de entrada ingresado.
- **def decToRoman(normal):** Realiza la conversión de decimal a romano.

Capturas del programa en ejecución







Hoja de Desempeño

Nombre	N° de Cuenta	% de Desempeño
Abner Eliezer Betancourt	20151021802	100
Andrés Fernando Lizardo Rodríguez	20161001202	100
Fidel Ernesto García Gutiérrez	20151001680	100
Jose Carlos Velásquez	20141031775	100
Roger Alfredo Molina	20161001463	100
Víctor Elías Vásquez Padilla	20161003401	100
Tomasa Clementina Meza Ochoa	20131000142	100

Bibliografía

- *Sistemas de Numeración.* (2021). Lógica de Programación.
<http://logicapusta.blogspot.com/2016/02/sistemas-de-numeracion.html>
- *SuperProf.* (2021). Sistema de numeración romana.
<https://www.superprof.es/diccionario/matematicas/aritmetica/numeros-romanos.html>
- *Regular Expression HOWTO — Python 3.10.0 documentation.* (2021).
Documentation Python. <https://docs.python.org/3/howto/regex.html>
- *¿Qué es un Analizador Sintáctico? - Ryte Marketing Wiki.* (s. f.). Analizador Sintáctico. https://es.ryte.com/wiki/Analizador_Sint%C3%A1ctico