



**INSTITUTO TECNOLOGICO DE LA LAGUNA**

**METODOS NUMERICOS**

**ACTIVIDAD DE APRENDIZAJE GRUPAL #1**

**TEMA 1: INTRODUCCION A LOS METODOS NUMERICOS**

**Número de control – Nombre del alumno:**

**16131309 – LUEVANO VILLANUEVA HECTOR MANUEL**

**17130010 – BURCIAGA IBAÑEZ MARCO ANTONIO**

**17130028 – GARCIA ALMANZA PATRICIA**

**18131069 – SAUCEDO SOSA DANIEL**

**18131209 – ADAME SANDOVAL JOSE MISAEL**

**Docente:**

**M. C. EDGAR ANTONIO PEÑA DOMINGUEZ**

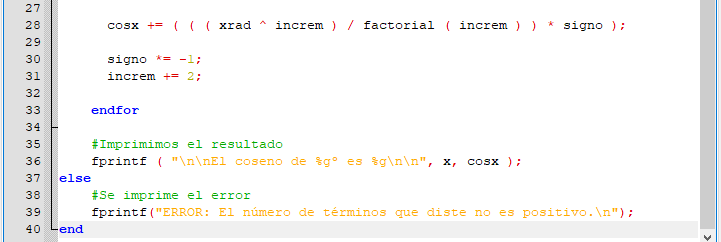
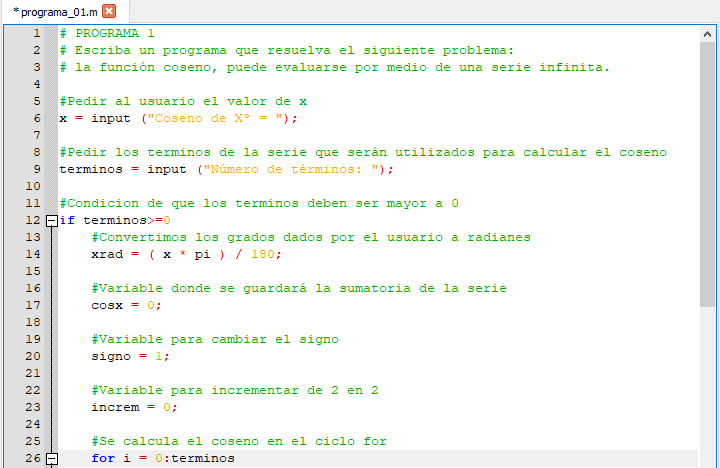
**Torreón, Coahuila 24 de junio del 2019**

**PROBLEMA 1**

Escriba un programa en Octave que resuelva el siguiente problema: la función coseno, puede evaluarse por medio de una serie infinita como:

El programa debe implementar la fórmula de modo que calcule e imprima los valores de conforme se agregue cada término de la serie.

**Código**



**Descripción**

Se pide al usuario que ingrese el valor de x en grados sexagesimales usando input

Se utiliza la condicional if para determinar si el término ingresado es mayor de 0 (ya que no existen términos negativos), si cumple la condición entonces hace el cálculo de la serie del coseno, sino imprime en pantalla un mensaje de Error y termina el programa.

Antes de hacer el cálculo de la serie se procede a hacer la conversión de grados sexagesimales a radianes, debido a que esta serie trabaja con radianes, para ello se declara la variable xrad.

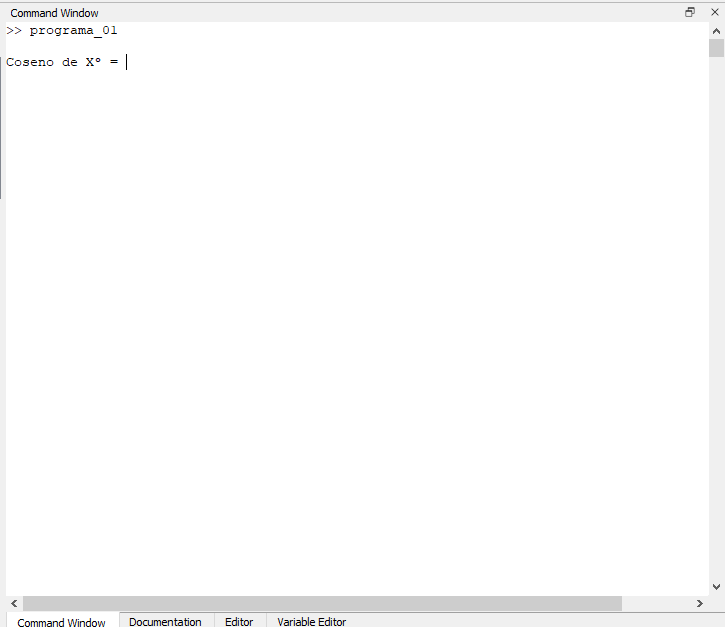
Después, se declaran tres variables cosx = 0, signo = 1 e incremen = 0

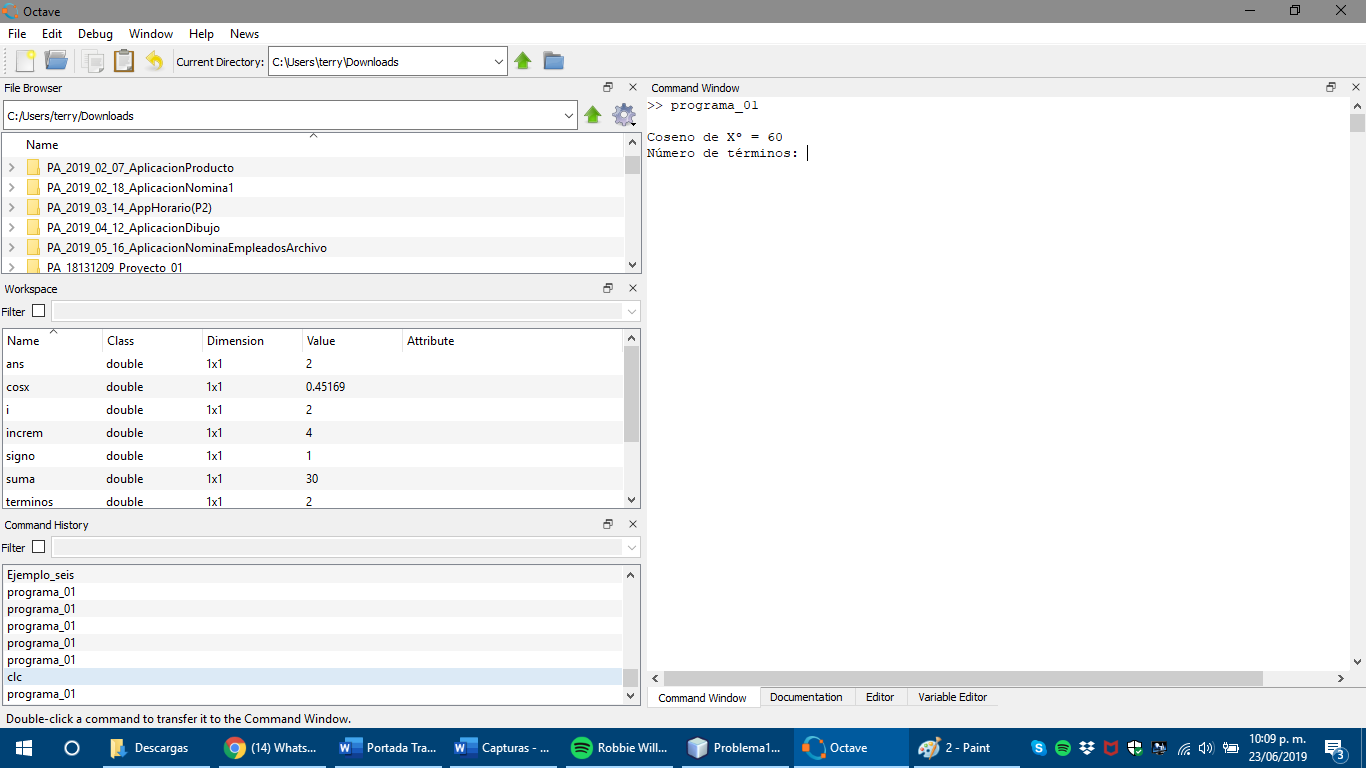
Ahora se realiza el cálculo (como se observa en la captura de pantalla) usando el ciclo for dónde el contador i = 1 hasta el término que ingreso el usuario, y se incrementará el contador de uno en uno, y en donde la variable cosx se estará sumando, la variable signo se estará multiplicando por – ya que es una serie alterna y la variable incremen se va a incrementar de dos en dos

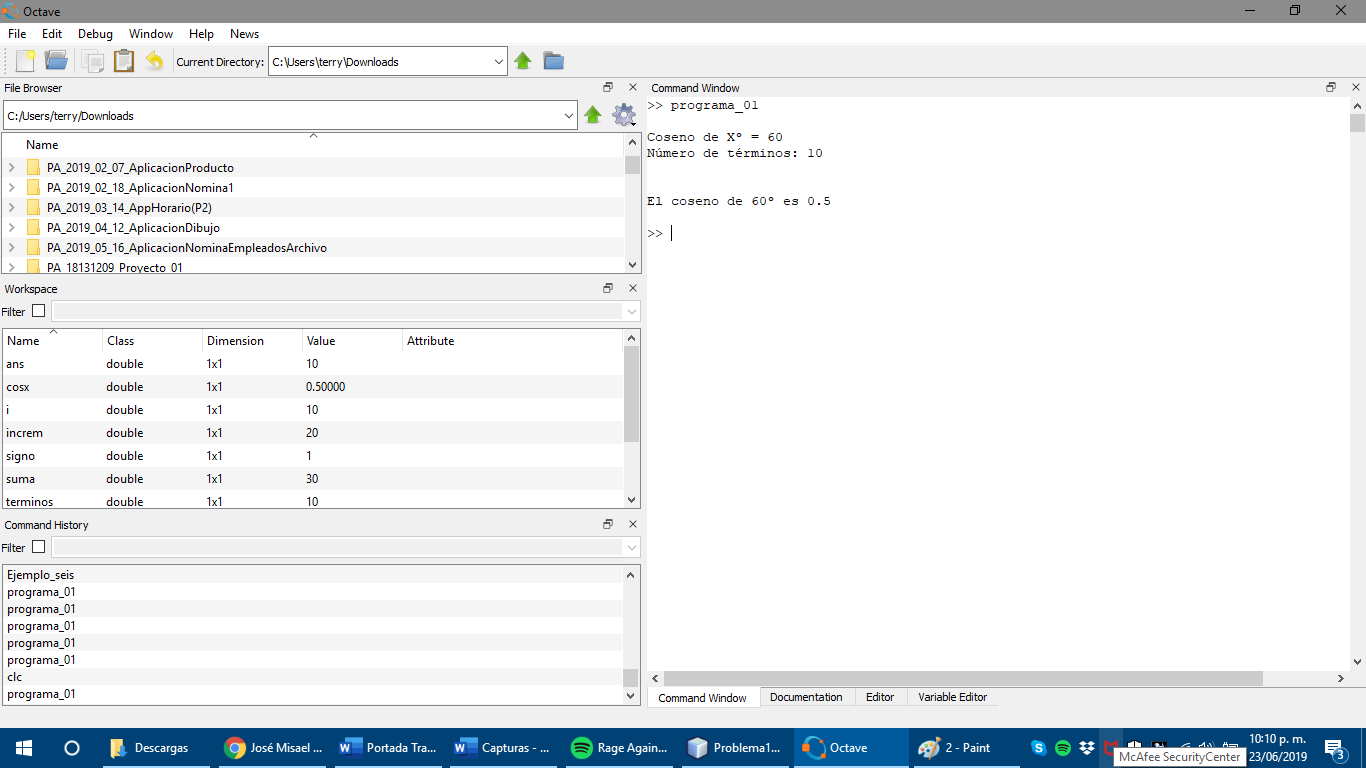
Al finalizar el ciclo for, se muestra el resultado de la sumatoria de la serie y termina el programa.

**Ejecución**

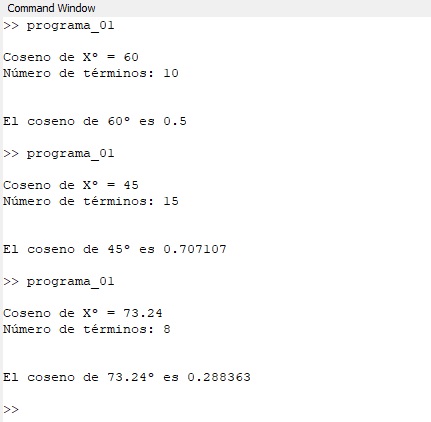
(paso a paso)







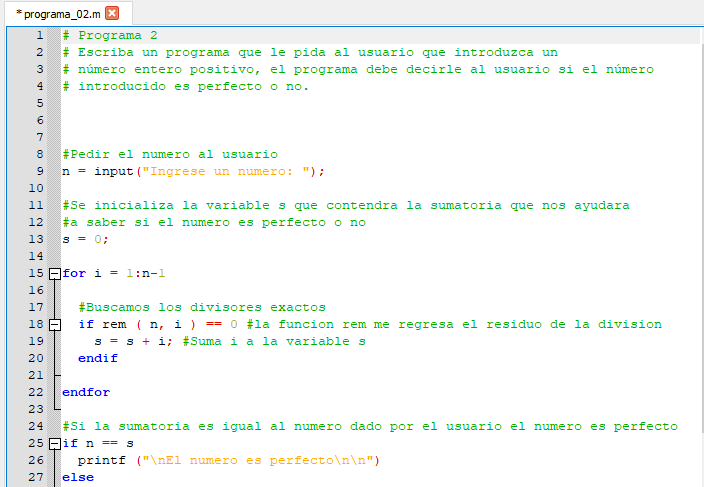
Ejecutado tres veces



**PROBLEMA 2**

Escriba un programa en Octave que le pida al usuario que introduzca un número entero positivo, el programa debe decirle al usuario si el número introducido es perfecto o no. Para complementar se sabe que un número perfecto es aquel que resulta de la suma de sus divisores exactos, como el 6, sus divisores exactos son: el 1, el 2 y el 3, al sumarlos dan como resultado el 6.

**Código**





**Descripción**

Se pide al usuario que ingrese un numero entero x usando input.

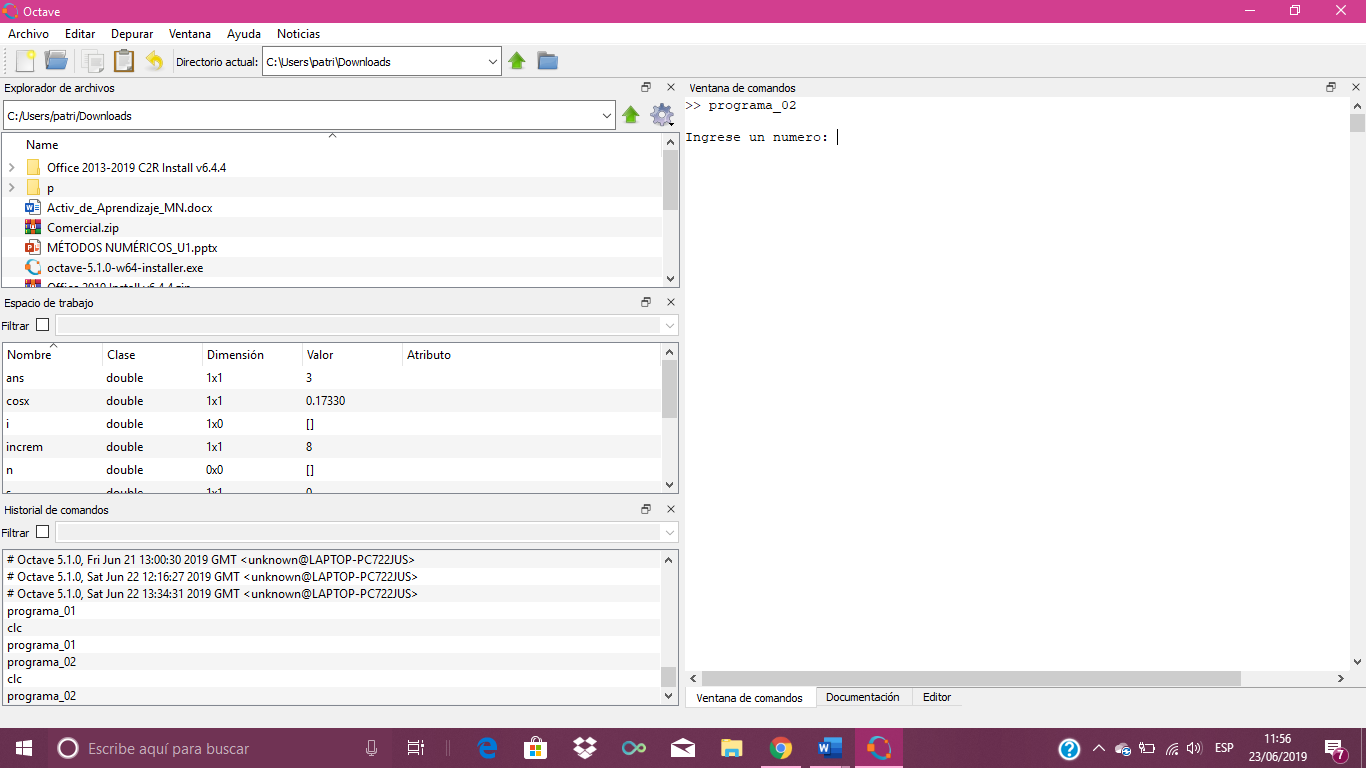
Después se inicializa la variable s en 0 que va a contener la sumatoria para determinar si el número es perfecto o no.

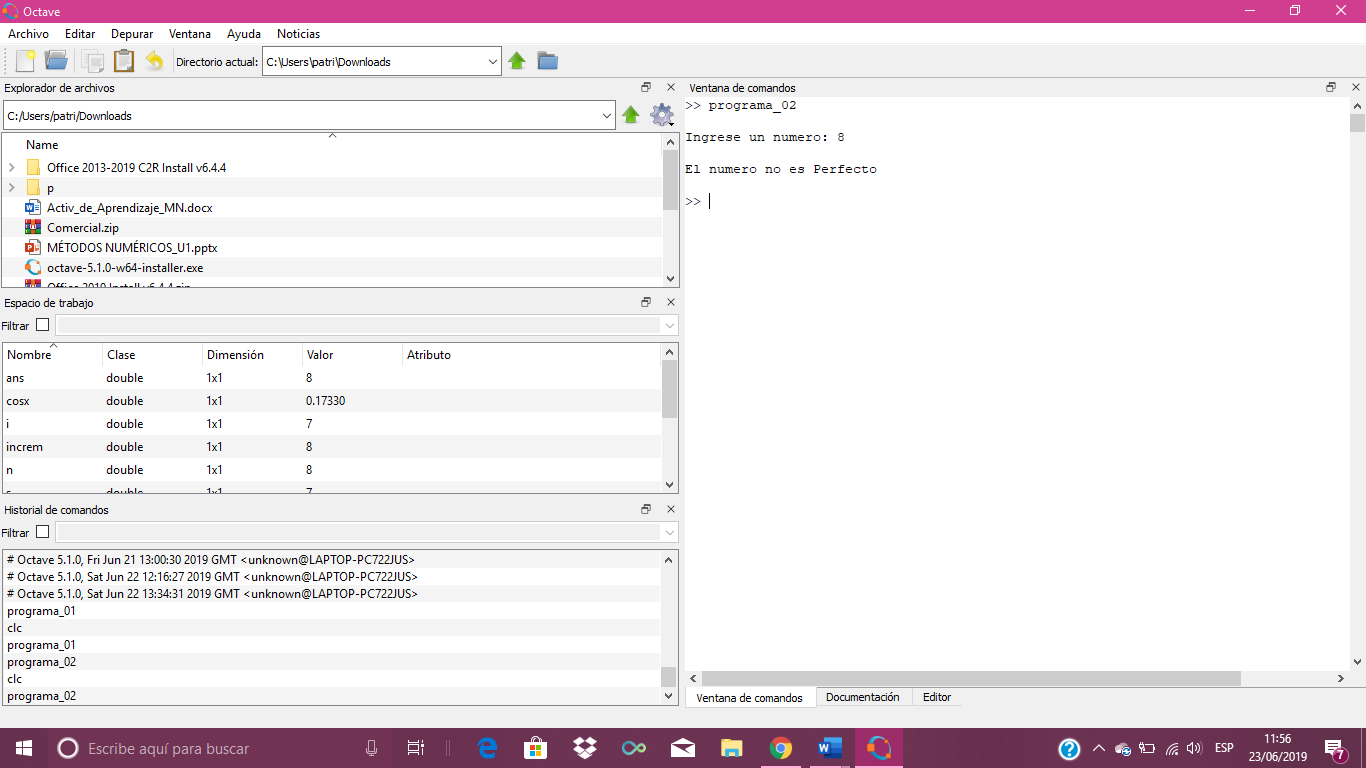
Ahora usando el ciclo for dónde el contador i = 1 hasta el número y restando uno que ingreso el usuario, y se incrementará el contador de uno en uno, dentro de este habrá un condicional if para buscar los divisores exactos usando la función rem que es la que regresa la división (rem = 0), si cumple la condición entonces suma i a la variable s, sin embargo termina el ciclo for cuando no se cumple la condición if.

Se vuelve a poner otra condición if para determinar si la sumatoria realizada es igual al número que ingreso el usuario, si cumple la condición entonces muestra el mensaje de que es un número perfecto y termina el programa, sino la cumple se procede a mostrar el mensaje de que no es un número perfecto y se finaliza el programa.

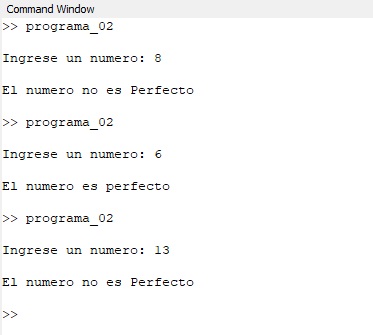
**Ejecución**

Paso por paso





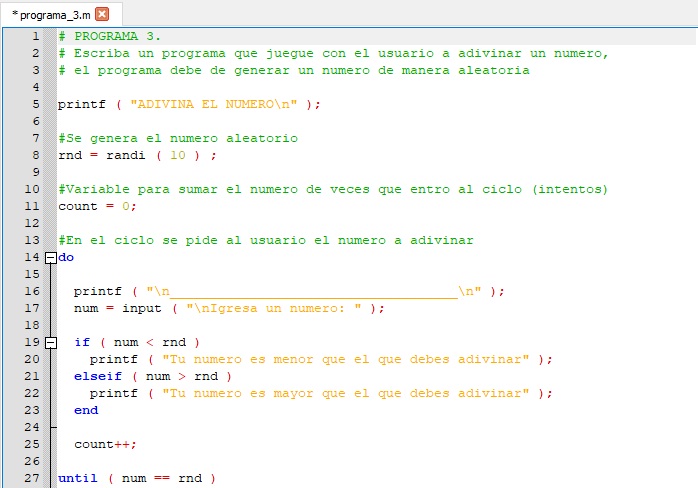
Ejecutado tres veces

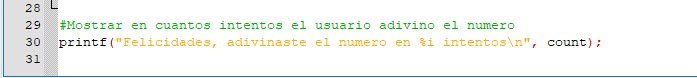


**PROBLEMA 3**

Escriba un programa en Octave que juegue con el usuario a adivinar un número, el programa debe de generar un número de manera aleatoria, el usuario intentará adivinar el número generado por el programa y este le debe indicar si el número que está introduciendo es mayor o menor al que está tratando de adivinar, una vez que el usuario de con el número, el programa le debe indicar que ha logrado adivinar el número además del número de intentos en el que lo logró.

**Código**





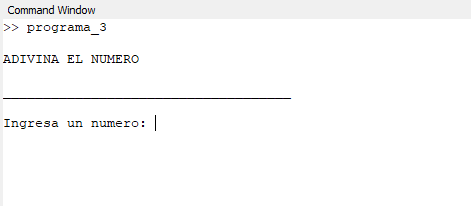
**Descripción**

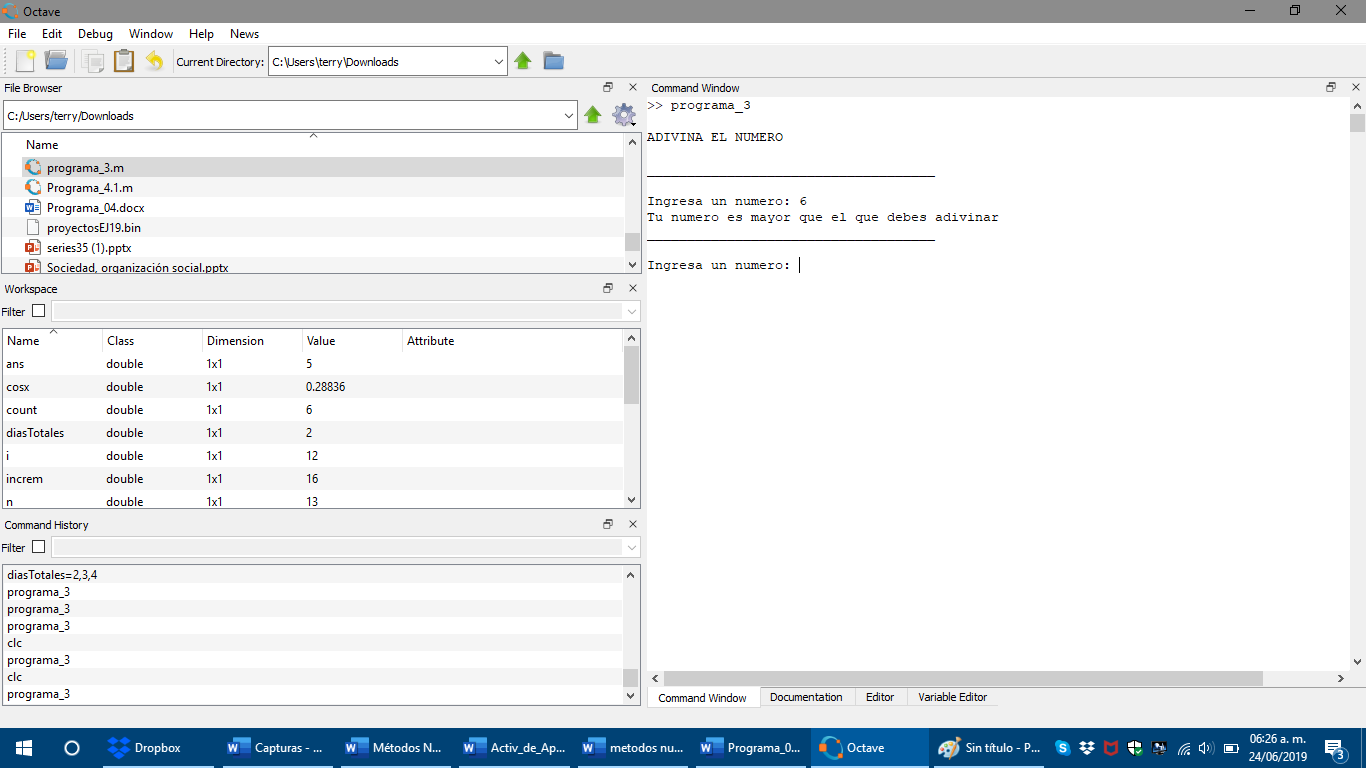
Se pone el título del programa, después se declara una variable rnd para guardar un número aleatorio de 0 a 10 el cuál el usuario tratará de adivinar, luego se declara otra variable llamada count la cual se inicializa en 0.

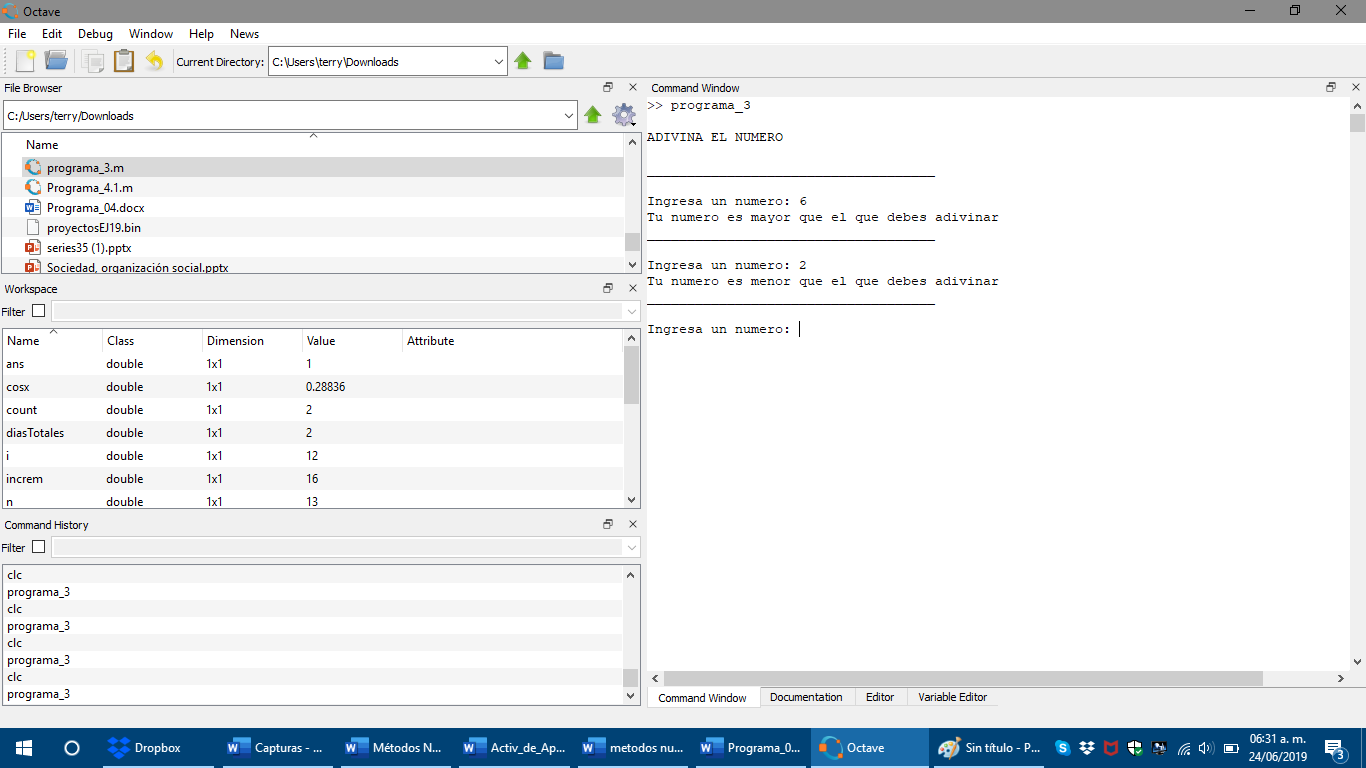
Usando el ciclo do until en donde entra por lo menos una vez hasta que el usuario adivine el número aleatorio (sin importar el número de intentos), dentro de este se pide al usuario que trate de adivinar el numero aleatorio (usando input), posteriormente se usa el condicional if para determinar que si el número que ingreso el usuario sea menor al que se trata de adivinar entonces se imprime de que efectivamente es menor, en caso de que no se cumpla se crea otro if para comprobar que si el número que ingreso el usuario sea mayor al que trata de adivinar entonces se muestra en pantalla que sigue siendo mayor, sin embargo, cuando no se cumplen ninguna de las dos condiciones entonces se suma 1 la variable count que es el número de intentos, finalmente se llega al until en donde se analiza si ya adivino el número el usuario, en caso de que no, se repite el ciclo, y por el contrario cuando es si, se finaliza el ciclo, por lo tanto, se imprimirá el mensaje de que ya adivinó el número con su número de intentos.

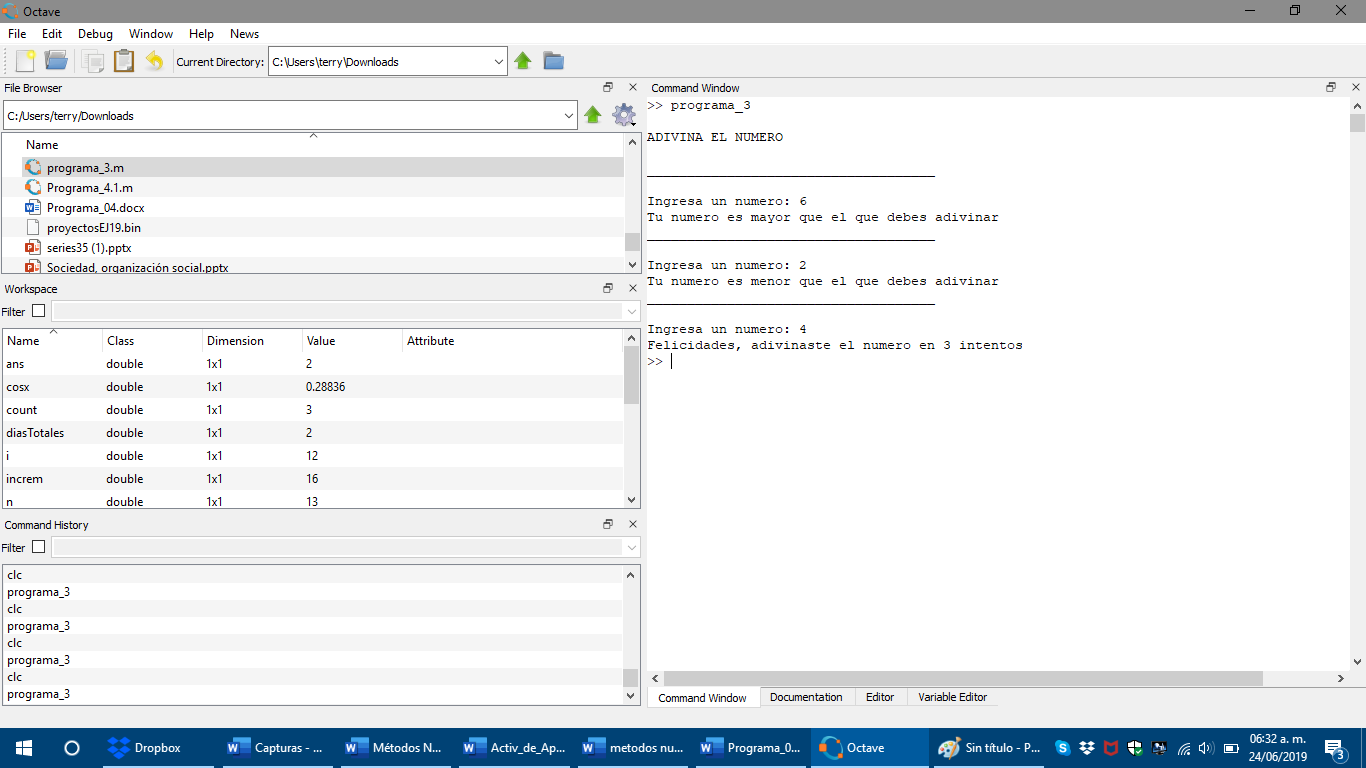
**Ejecución**

Paso por paso

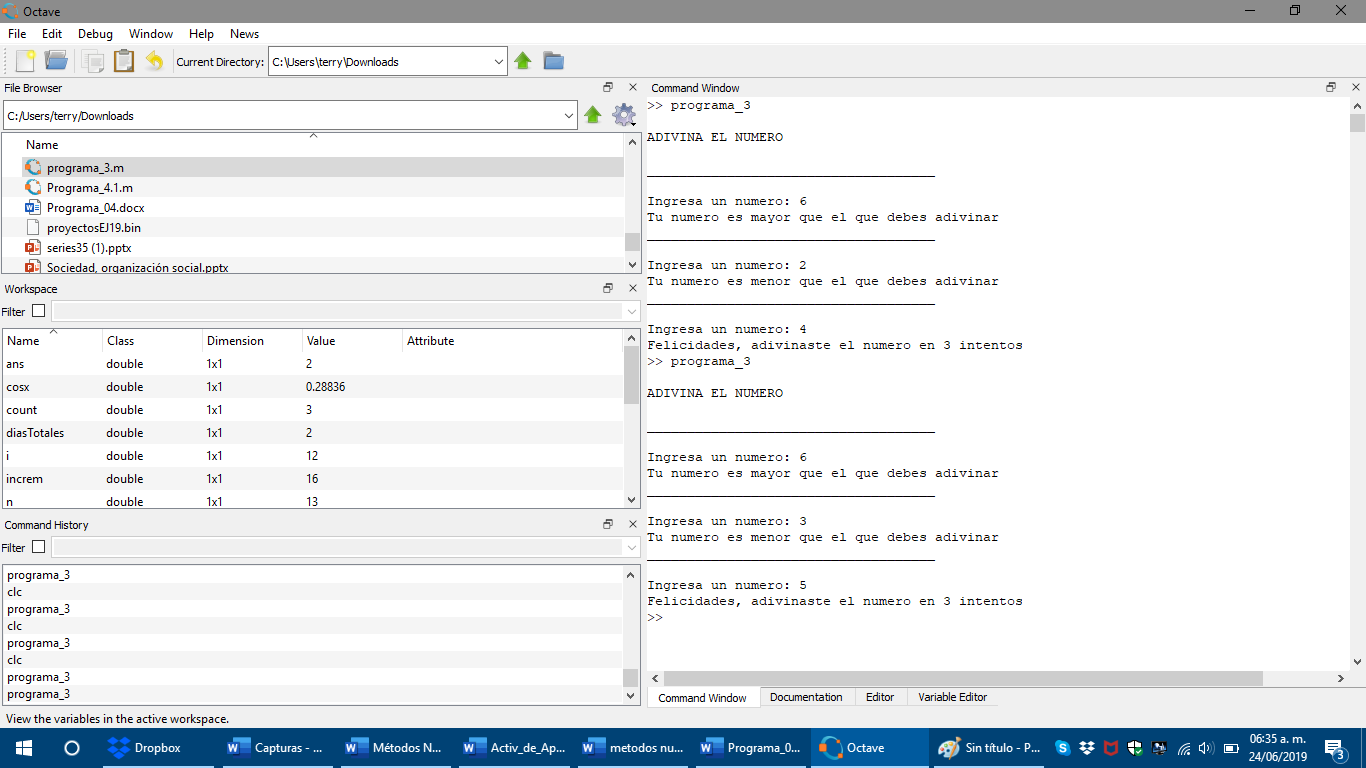


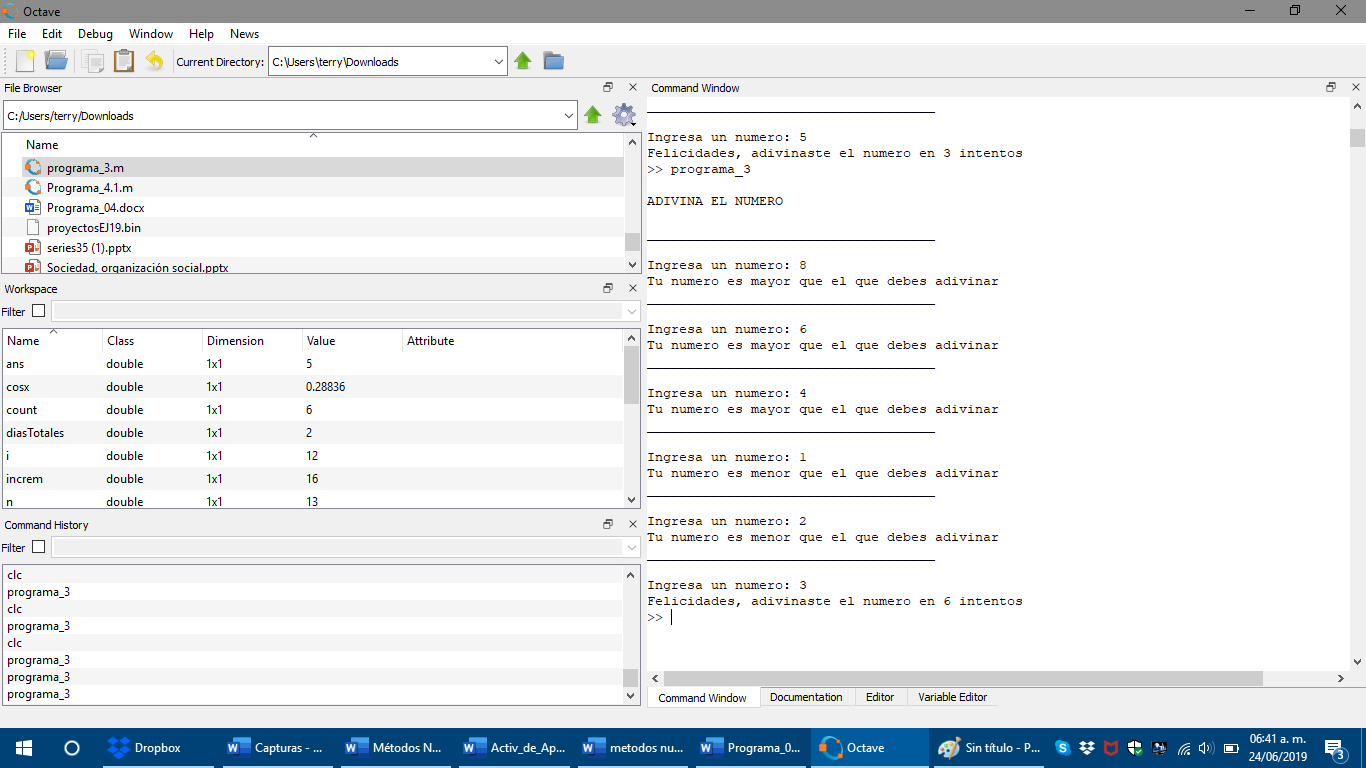






Ejecutado dos veces

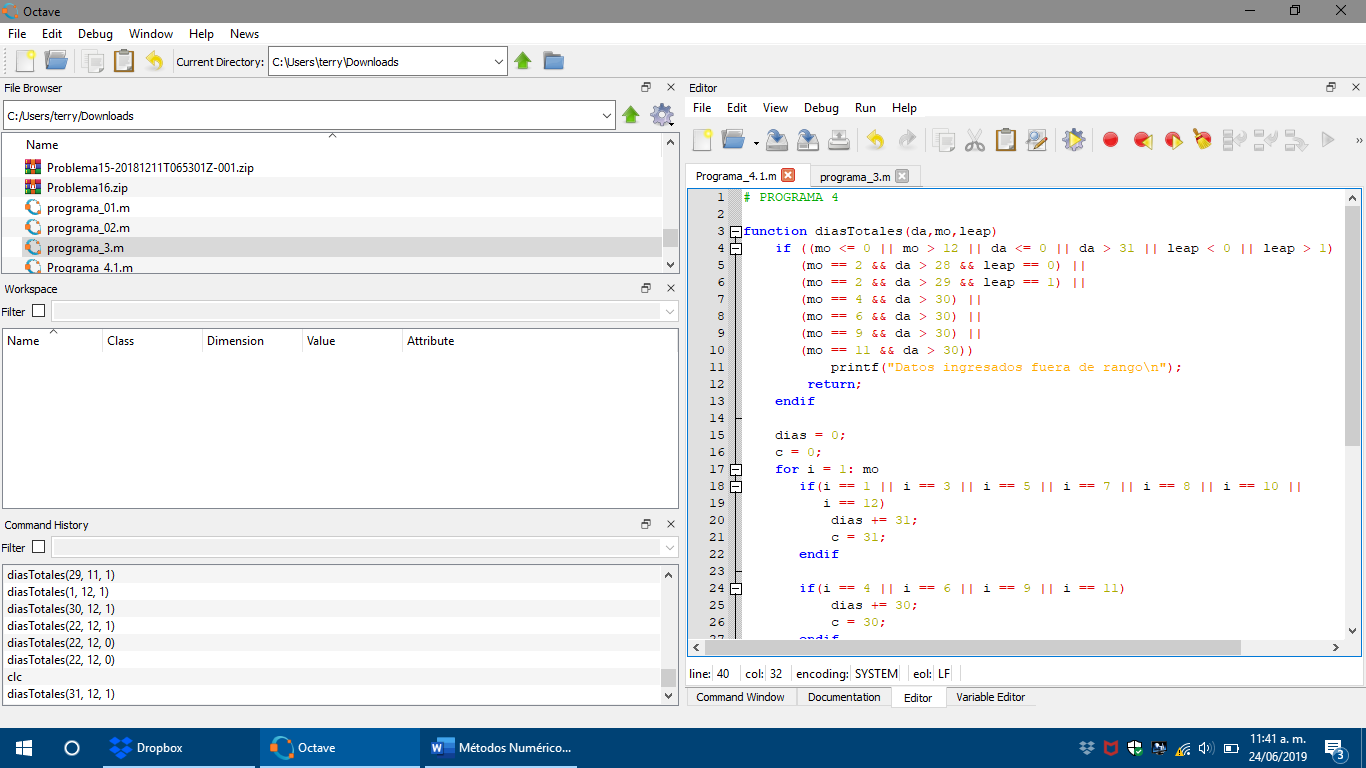


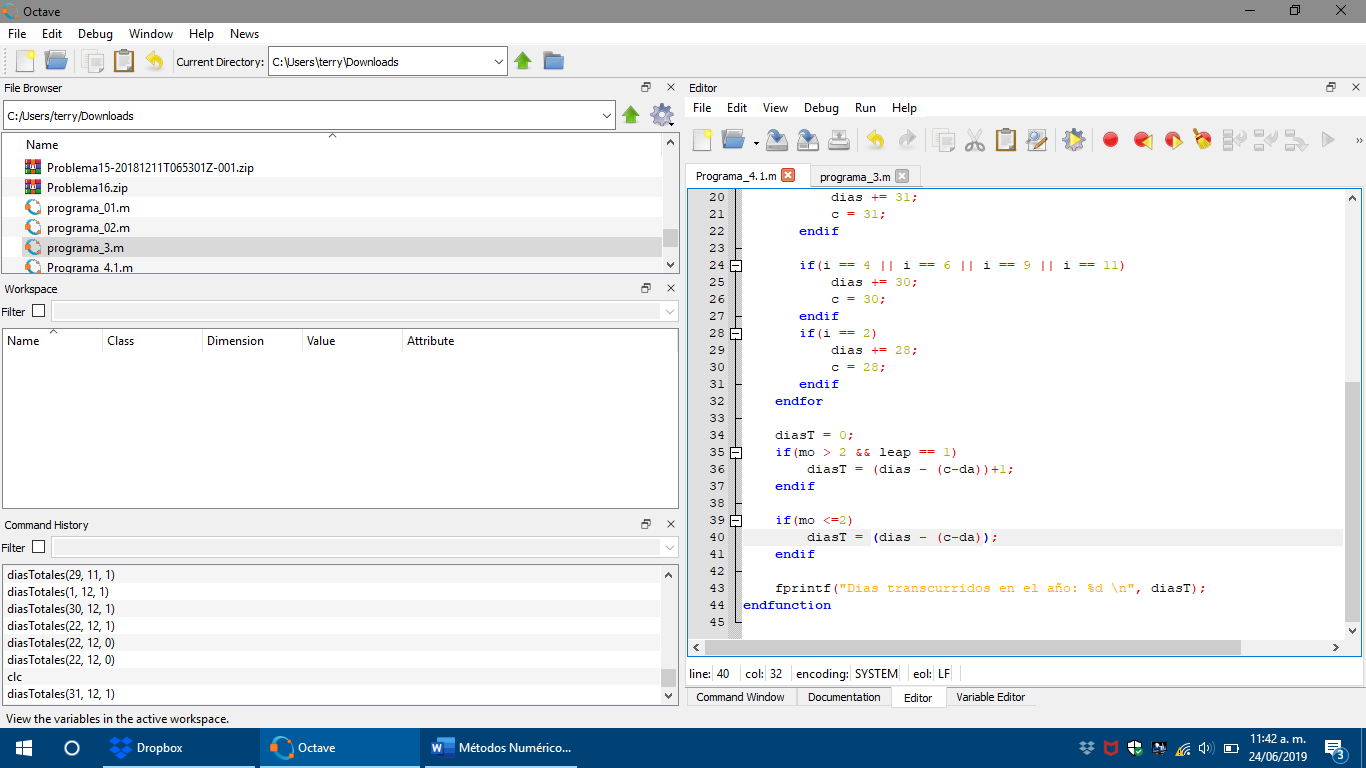


**PROBLEMA 4**

Desarrolle una función en Octave para determinar los días transcurridos en un año. La función debe pasar por tres valores: mo = el mes (1-12), da = la fecha (1-31), y leap (bisiesto) (0 para año no bisiesto y 1 para año bisiesto). Pruébela para 1 de Enero de 1999, 29 de Febrero del 2000 y 31 de Diciembre del 2004.

**Código**





**Descripción**

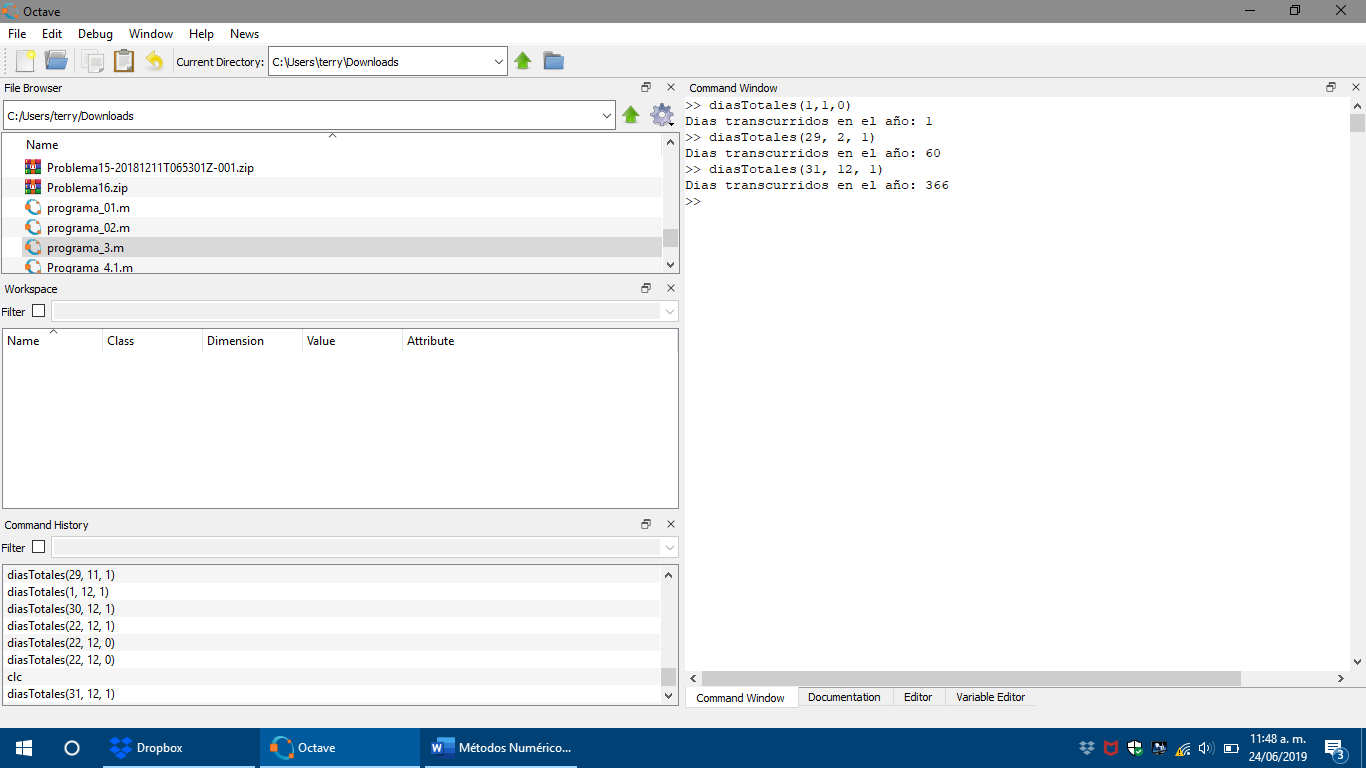
Primero se crea la función con function, en dónde el usuario ingresará números enteros a las variables da, mo y leap (días, meses y si es bisiesto el año)

Se hace una condicional if para desechar días existentes como el 30 de febrero, 31 de Junio, entre otros, además de que sólo hay año normal y bisiesto (que es un día más), en caso de cumplir esta condición se muestra un mensaje de que son erróneos los datos y se finaliza el programa.

Se inicializan dos variables c y días en 0, posteriormente con un ciclo for donde i = 1 hasta el mes que ingresó el usuario, dentro de este se hacen tres condicionales if en donde se checan los meses con 31, 30 o 28 días, y pues al cumplir la condición se van sumando respectivamente los días (representado con una i en el ciclo) y el c que es el contador se actualiza al mes que ingresa el usuario (servirá para restar los días).

Luego, como el año bisiesto en el mes de febrero son 29 días se hace una condición if donde el mes sea mayor que 2 (febrero) para agregar el día extra y calcular los días transcurridos en el año, sino se cumple, solamente calcula los días transcurridos.

**Ejecución**



31 de Diciembre del 2004 1999

29 de Febrero del 1999

1 de Enero de 1999