

MATLAB y la teoría de números

David Báez López

En este capítulo se han incluido los siguientes simuladores desarrollados con MATLAB:

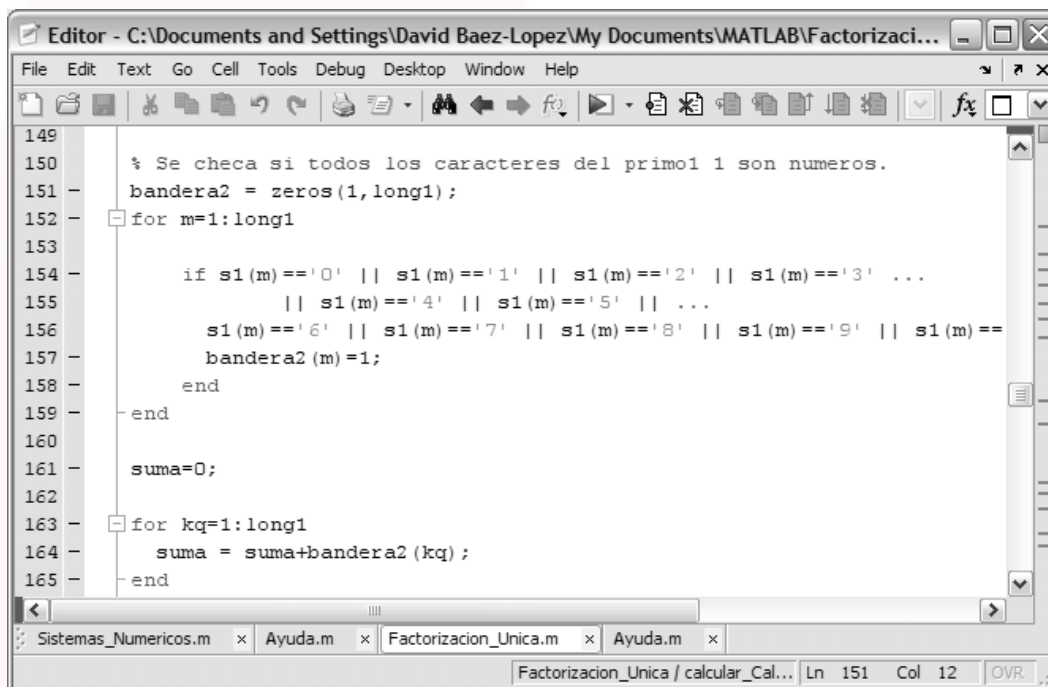
- **Números primos**
- **Sistemas numéricos**
- **Máximo común divisor**
- **Factorización única**
- **Calendario perpetuo**

Esta es una muestra de lo que se puede hacer con MATLAB al aplicarlo a la teoría de números

Los simuladores **Factorización única** y **Sistemas numéricos** de este capítulo se realizaron usando el lenguaje m de MATLAB y la herramienta GUIDE para generar la interfaz gráfica para la entrada y despliegue de datos.

Las instrucciones más usadas en estos simuladores son las instrucciones if-end y bucles for.

En general, usando la instrucción if-end se toman decisiones para poder realizar las operaciones deseadas. Por ejemplo, en relación con el simulador **Sistemas numéricos**, la siguiente figura muestra las decisiones tomadas para saber si los caracteres de un número dado al simulador son caracteres numéricos:



```
149
150 % Se checa si todos los caracteres del primo1 1 son numeros.
151 bandera2 = zeros(1,long1);
152 for m=1:long1
153
154     if s1(m)=='0' || s1(m)=='1' || s1(m)=='2' || s1(m)=='3' ...
155         || s1(m)=='4' || s1(m)=='5' || ...
156         s1(m)=='6' || s1(m)=='7' || s1(m)=='8' || s1(m)=='9' || s1(m)=='
157         bandera2(m)=1;
158     end
159 end
160
161 suma=0;
162
163 for kq=1:long1
164     suma = suma+bandera2(kq);
165 end
```

En esta figura se ve que dentro de los bucles for se incluyen instrucciones if que permiten tomar decisiones dentro de los mismos bucles.

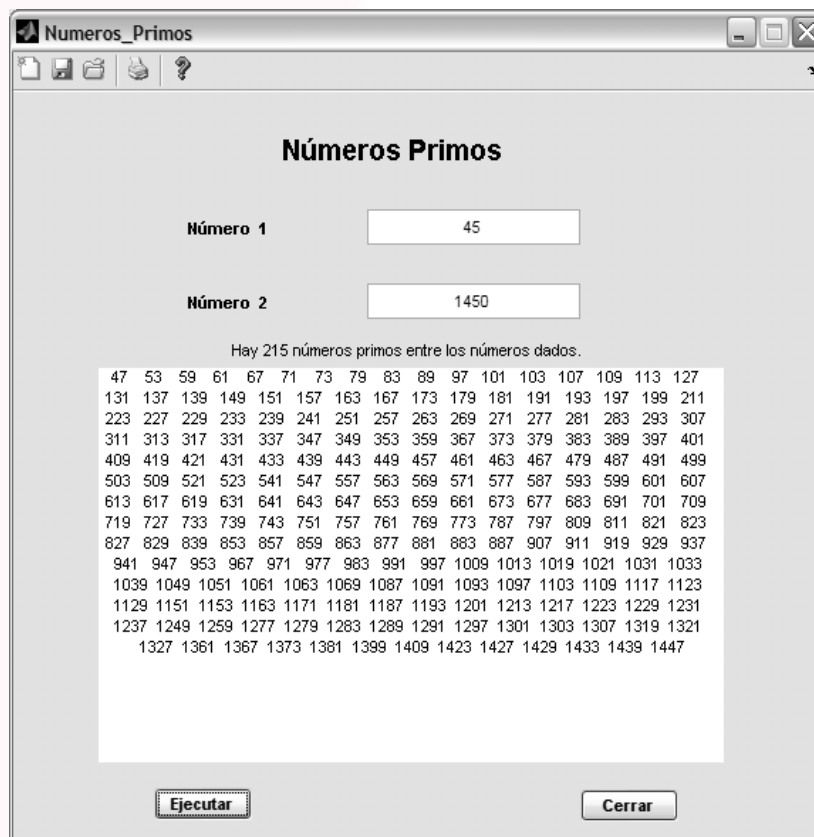
En el capítulo 7 del libro **MATLAB con aplicaciones a la Ingeniería, Física y Finanzas**, 2ª edición, escrito por el Dr. David Báez López y publicado por Alfaomega Grupo Editor, se describen detalladamente y se ejemplifican todas las instrucciones incluidas en la programación de estos simuladores mientras que en el capítulo 9 se expone la estructura general de la herramienta GUIDE y el procedimiento para implementar una GUI específica.

Números primos

El propósito general de este simulador es calcular los números primos que hay entre dos números enteros arbitrarios en el intervalo $(1, 2^{32})$.

Además de instrucciones if-end y bucles for, se usa la instrucción isprime para verificar si un número dado

es un número primo. Este simulador cuenta con 463 líneas de código y la siguiente figura muestra la GUI del mismo:



Sistemas Numéricos

Este simulador usa las instrucciones `if` y `for` para leer y validar números y sus bases para realizar conversiones de ellos a otras bases así como realizar operaciones con ellos. Las operaciones que se pueden realizar son la suma la resta y la multiplicación. Las bases posibles son desde base binaria a hexadecimal. Algunas de las bases vienen predeterminadas y con solo seleccionarlás el simulador ejecuta la conversión. También se puede realizar la conversión con una base arbitraria. La siguiente figura ilustra el cálculo de un producto en base 8 de tres números de base arbitraria.

Sistemas Numéricos

Números: 2(9), 45(11), 23(8)

Resultado: 304

Convertir de decimal a binario Convertir de binario a decimal

Convertir de decimal a hexadecimal Convertir de hexadecimal a decimal

Convertir de decimal a octal Convertir de octal a decimal

Conversión de decimal a base [] Conversión de base [] a decimal

Suma en base [] Resta en base [] **Producto en base** [8]

Limpiar Cerrar

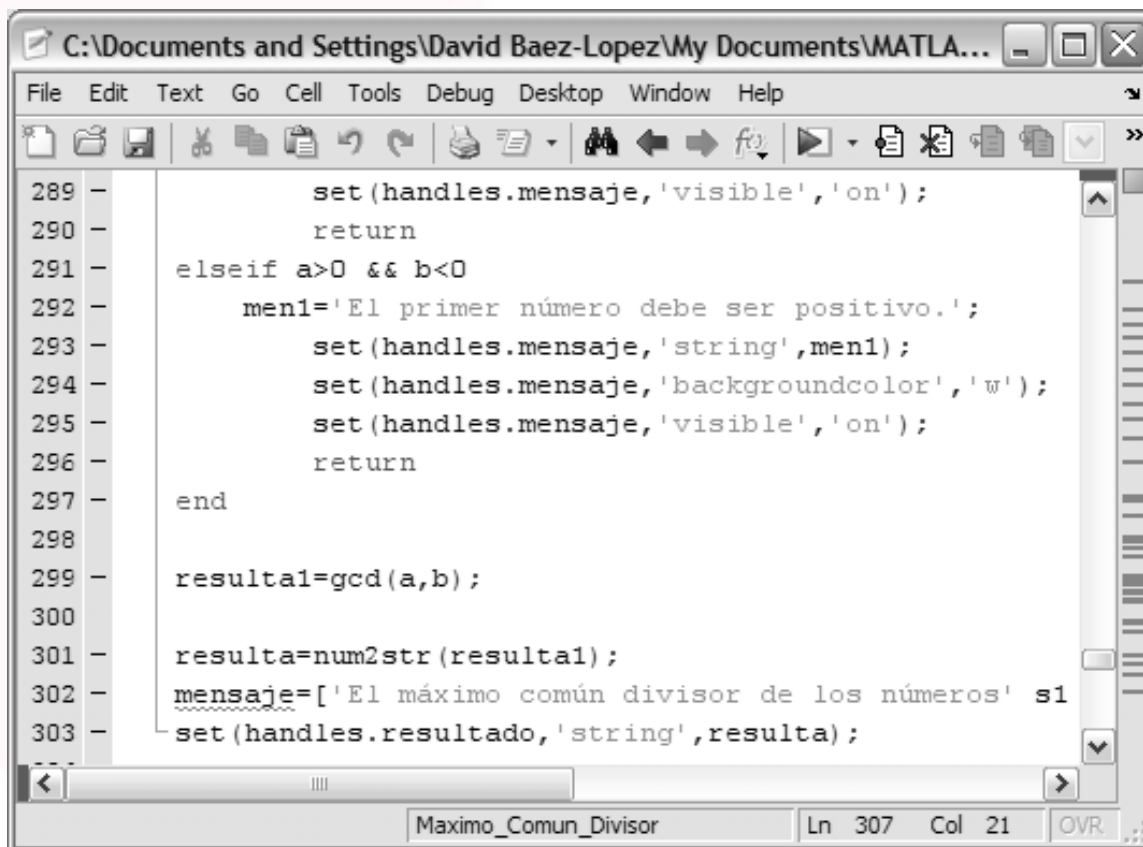
Este simulador tiene una ayuda que indica como introducir los datos al simulador. El simulador Sistemas numéricos tiene 6334 líneas de código.

Factorización Única

Este simulador usa las instrucciones plot, if y for para graficar la posición y tamaño de los círculos así como sombrear con asteriscos la parte de los círculos, ya sea dentro o fuera, de acuerdo a la función escogida del menú presentado en la interface gráfica. Esta interface tiene 1503 líneas de código.

Máximo Común Divisor

Este simulador determina el máximo común divisor (MCD) de dos números usando el algoritmo de Euclides. Este algoritmo está programado en la función de MATLAB gcd que son las iniciales en inglés de greatest common divisor (máximo común divisor). Además para la validación de los datos de entrada se usan las instrucciones if-end y bucles for. La siguiente figura ilustra la parte del código donde se calcula el MCD:

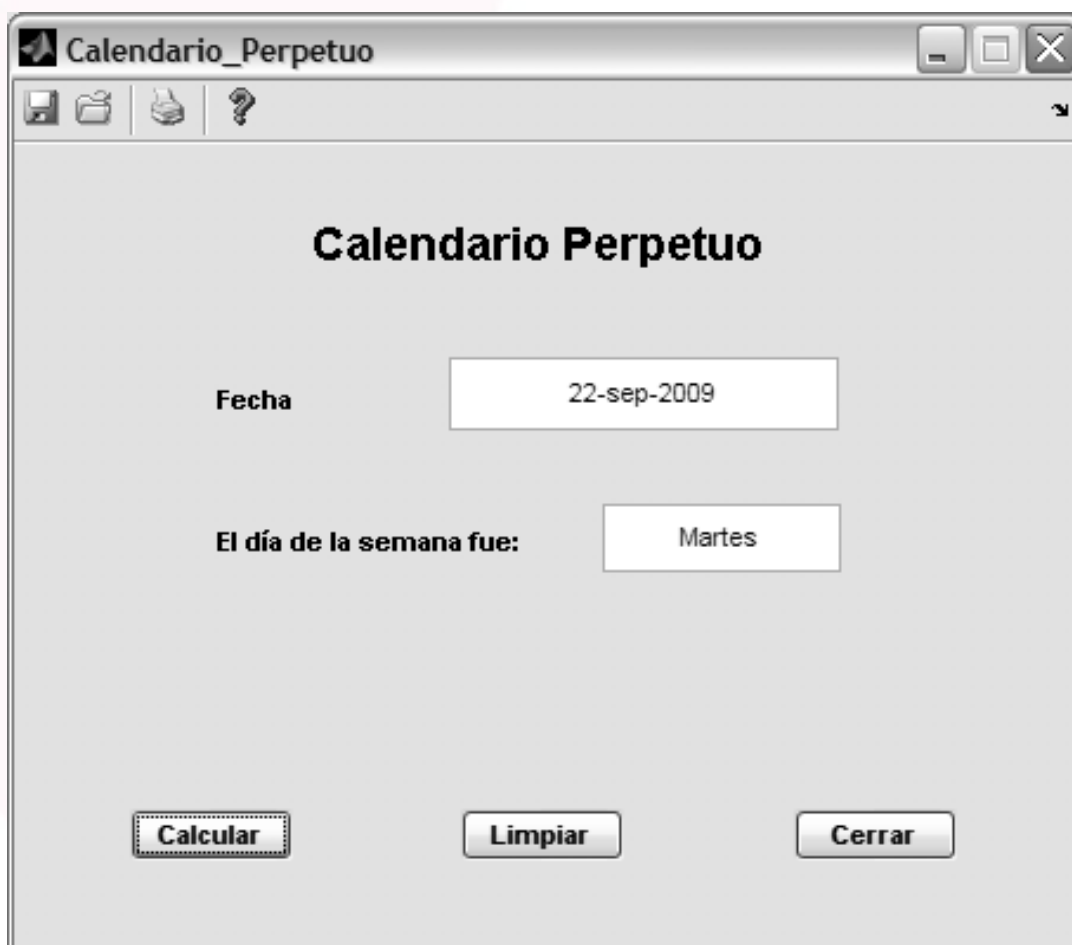


```
289 -         set(handles.mensaje,'visible','on');
290 -         return
291 -     elseif a>0 && b<0
292 -         men1='El primer número debe ser positivo.';
293 -         set(handles.mensaje,'string',men1);
294 -         set(handles.mensaje,'backgroundcolor','w');
295 -         set(handles.mensaje,'visible','on');
296 -         return
297 -     end
298
299 -     resulta1=gcd(a,b);
300
301 -     resulta=num2str(resulta1);
302 -     mensaje=['El máximo común divisor de los números' s1
303 -     set(handles.resultado,'string',resulta);
```

El simulador tiene un total de 413 líneas.

Calendario Perpetuo

Este simulador permite calcular el día de la semana de una fecha cualquiera después del 1 de enero de 1600. El formato de entrada de datos es día-mes-año. La ayuda proporcionada por este simulador permite ver los distintos formatos de entrada de la fecha. La figura siguiente muestra un ejemplo del uso del simulador para la fecha 22 de septiembre del año 2009.



Se usan instrucciones if-elseif-end y bucles for, junto con la herramienta de desarrollo de interfaces gráficas GUIDE para desarrollar este simulador. La siguiente figura muestra parte del código que tiene un total de 449 líneas de código.