Apellidos y Nombre:
Grupo : 3A

INTRODUCCION A MATHEMATICA

¿Qué es *Mathematica*?

Mathematica es un paquete de software de Matemáticas de uso fácil y gran capacidad que integra los elementos siguientes:
Cálculo numérico, Cálculo simbólico, Grafismo en 2D/3D, Lenguaje de programación y Aplicaciones
En esta introducción nos iniciaremos en el conocimiento básico de las funciones más importantes de Mathematica que nos serán imprescindibles para el posterior desarrollo de las prácticas de las diversas asignaturas de matemáticas que han sido cursadas hasta el momento. En cada una de las prácticas posteriores se hará un estudio de los comandos específicos que son necesarios para el desarrollo de las mismas.

Estructura interna de Mathematica

Este paquete tiene una estructura que permite su utilización en diferentes entornos: Pc, Mac, Unix, etc.
Para ello está dispuesto en dos partes fundamentales. El Kernel (motor de cálculo) y el Front End (Interfaz de comunicación con el usuario). Cuando se comienza a utilizar el programa se activa el Front End con lo que se puede comenzar a introducir datos y expresiones. En el momento en que se desea realizar la primera operación se activa el Kernel que es el módulo que realiza el cálculo. En el Kernel se encuentran introducidos los procedimientos de cálculo relacionados con una gran cantidad de operaciones (las más habituales). Sin embargo, dada la potencia de cálculo del paquete, operaciones y procedimientos más complejos se encuentran almacenados en diferentes "Packages" que deben ser activados antes de realizar los cálculos con ese tipo de sentencias.
En el entorno Windows en que nos movemos, el Front End está dispuesto de manera que todo el desarrollo realizado puede ser almacenado prácticamente como en un editor de texto y tiene además todas las características de conectividad con otros paquetes que funcionen en el mismo entorno windows .
También, muchos de los comandos que ejecuta *Mathematica* han sido introducidos en ventanas gráficas "Palettes" que permiten una más rápida implementación de las sentencias y evitan memorizar parte de los comandos utilizados.
Todas estas características permiten un manejo sencillo y rápido de modo que, con una pequeña introducción a las funciones básicas, al modo de introducir datos y a la utilización de la ayuda, se puede manejar con soltura en un corto espacio de tiempo.

Empezando a trabajar con Mathematica

Entradas/salidas, celdas, %, %n

Comenzar a trabajar con Mathematica es muy sencillo, basta con introducir la operación que se desea realizar y pulsar las teclas Shift + Intro (o también la tecla Intro del teclado numérico). Así Mathematica realiza la operación indicada y genera una serie de celdas de entrada y salida perfectamente distinguibles que podremos manipular como cualquier objeto del entorno Windows.
Procediendo de este modo las operaciones quedan numeradas en el orden en que se van realizando, con lo que se pueden ir utilizando los resultados previos con solo indicar en qué momento se obtuvieron. El resultado de la última operación se puede recuperar utilizando el símbolo %, el penúltimo mediante %% y en general, el resultado de la k-ésima operación con el símbolo %k.

```
2 * 3
```

$$3+\left(\frac{5}{2}\right)^2$$

```
2.15 - (3 / 4) * 2
```

```
% - %1
```

La barra de menús

De la barra de menús distinguiremos especialmente las siguientes partes: File, Edit, Kernel y el Help. En ellas se encuentran recogidas aquellas acciones de uso más frecuente: abrir, salvar, imprimir, copiar, pegar, salirse del Kernel, etc.
Comentar forma rápida de copiar una celda
El ágil manejo de la ayuda nos será de especial utilidad en nuestro trabajo con *Mathematica*
Aquí ver por encima todas las opciones de la barra y con todo detalle Help

Números y operaciones

Números: racionales, con punto decimal y complejos

En *Mathematica* se puede trabajar con números racionales, con punto decimal o complejos, expresados estos últimos en la forma x+yI

Operaciones aritméticas: + , - , * , / ,^

Las operaciones aritméticas: adición (+), sustracción (-), multiplicación (*), división (/) y potenciación (^) nos permiten trabajar con números . El orden jerárquico de las mismas es el usual, empleándose paréntesis (nunca corchetes) para indicar las agrupaciones que interese realizar. El símbolo del producto (*) puede ser sustituído por un espacio en blanco

- **Números exactos y aproximados. N[exp] (o exp //N) , N[exp,n] , Chop[exp] (o exp//Chop), Chop[exp,tol]**

Mathematica siempre que puede llega al resultado exacto. Ahora bien, en ocasiones puede interesar llegar a un resultado aproximado en vez de al resultado exacto, para hacerse una idea del orden de magnitud del resultado obtenido. Esto se puede hacer:

a) Introduciendo alguno de los factores de la operación que se desea realizar de forma aproximada. Para ello basta con poner en forma decimal algún factor.
b) Utilizando el comando N. Con él podemos incluso indicar el número de decimales que se desea aparezcan. Por defecto presenta 6 pero internamente trabaja con 16. Otra forma de utilizar el comando N es , al final //N
En ocasiones puede resultar de interés reemplazar numeros reales aproximados próximos a 0 por el número exacto 0. El comando Chop permite llevar a cabo tal reemplazamiento.

```
3 + (5 / 2) ^2
```

```
3. + (5 / 2) ^2
```

```
N[3 + (5 / 2) ^2]
```

```
3 + (5 / 2) ^2 // N
```

```
N[3^50]
```

```
N[3^50, 10]
```

```
InputForm[%]
```

```
Chop[2.54 10^-15]
```

```
Chop[2.54 10^-7, 10^-5]
```

Constantes y funciones predefinidas

- **Constantes predefinidas: Pi , E , Degree , I**

Mathematica dispone de ciertas constantes de interes matemático predefinidas. Las indicadas representan, respectivamente, a los números: π, e, π/180 y a la unidad imaginaria $\sqrt{-1}$
`Degree` para trabajar con grados en lugar de radianes.

- **Funciones elementales predefinidas: Sqr[_] , Exp[_] , Log[_] , Sin[_] , Cos[_] ,Tan[_] , ArcSin[_] , Abs[_] , ...**

Asímismo, *Mathematica* tiene predefinidas un amplio abanico de funciones elementales

Para una mayor comodidad, se puede recurrir a la utilización de paletas para introducir constantes y funciones predefinidas así como los comandos operativos más frecuentes.
Cualquier argumento entre corchetes. Los paréntesis se utilizan de la misma forma que en el papel o en la pizarra.

```
{Cos[60 Degree], Sin[90 °]}
```

Ejercicios

1- Utilice **N** para calcular π con 50 decimales.

2- Utilice N para ver a qué entero se aproxima E^{π}. A continuación utilizar los comandos Floor[E^{π}] y Ceiling[E^{π}], recurriendo previamente a la ayuda para obtener información acerca de cada uno de ellos.- Repetir con -E^{π}.

3- Calcule dos números aleatorios con Random[ ] y a continuación multiplíquelos
Vamos a la ayuda de Random (Random[],nº aleatorio entre 0 y 1, Random[Integer,{0,10}]:nº aleatorio entero entre 0 y 10.Random[Real,{0,100}]

Variables definidas por el usuario

- **Cómo escribir una variable. Operador asignación (=). Cómo pedir información sobre una variable: ? . Cómo borrar una variable: Clear[ ] , Remove[ ]**

Para definir una variable propia en *Mathematica* se puede utilizar cualquier sucesión alfanumérica que comience por una letra o el símbolo $. Puesto que las funciones internas de *Mathematica* siempre comienzan por una letra mayúscula, suele ser una práctica recomendable, en la definición de nuestras propias variables, comenzar por una letra minúscula.
Para asignar un valor a una variable se utiliza el operador asignación (=)
El comando Remove es más drástico que el comando Clear. Este borra valores y definiciones pero no atributos o mensajes ligados al simbolo. Remove borra todo.

```
x = 3
```

```
?x
```

```
x = 6
```

```
?x
```

```
Clear[x]
```

```
?x
```

```
Remove[x]
```

```
?x
```

Cálculos secuenciales

En *Mathematica* es posible realizar más de un cálculo en una misma entrada, siempre y cuando separemos los mismos por (;). Poniendo al final de una entrada (;) la salida correspondiente no se muestra en pantalla.

```
x = 3; y = -8; x^2 - y^3
```

```
x = 3; y = -8; x^2 - y^3;
```

```
Clear[x, y]
```

Cálculo simbólico. Operador reemplazamiento (/.)

Mathematica opera también con expresiones simbólicas. En una expresión simbólica podemos sustituir las variables por valores concretos, o bien por otras expresiones, mediante la utilización del operador reemplazamiento (/.).
Hay que destacar que este reemplazamiento es local, no quedando el valor asignado a dichas variables.

```
x^2 - 3 x + 1
```

```
x^3 - y^3 + x y
```

```
x = 1; y = 2;
```

```
x^3 - y^3 + x y
```

```
x^3 - y^3 + xy
```

```
Clear[x, y]
```

```
x^2 - 3 x + 1 /. x -> 1
```

```
x^2 - 3 x + 1 /. x -> a + b
```

```
x^3 - y^3 + x y /. {x -> 1, y -> 2}
```

```
? x
```

Listas. El comando Table.

Una lista permite representar conjuntos de elementos, los cuales puede que no sean del mismo tipo, como valores numéricos, variables, texto, etc. Para acceder a ciertos elementos(o a un elemento concreto) de una lista se considera la posición que ocupan en ella.

Por otra parte, las listas pueden ser manipuladas, en muchas ocasiones, como si de simples objetos se tratase.

Los comandos Dimensions y Length dan las dimensiones (para listas y matrices) , y el nº de elementos de la lista.

El comando Partition[lista,n] agrupa los elementos de la lista de n en n, crando una lista cuyos elementos son listas.

```
a = {1, -2, x, Sqr[2], 3 + y^2, x + y}
```

```
a[[3]]
```

Para seleccionar los elementos 1º, 3º y 5º

```
a[[{1, 3, 5}]]
```

Se suma -1 a todos los elementos de la lista y se halla el valor absoluto.

```
Abs[-1 + a]
```

```
Dimensions[a]
```

```
p = Partition[a, 2]
```

p es una lista de tres elementos que a su vez son listas de dos elementos: Matriz de tres filas y dos columnas. Lo comprobamos con el comando Dimensions

```
Dimensions[p]
```

```
q = Partition[a, 3]
```

```
Dimensions[q]
```

```
b = {{x, y, z}, {u, {v, w}, r}}
```

```
Dimensions[b]
```

Extrayendo elementos de b

```
b[[1, 2]]
```

```
b[[2, 2]]
```

```
b[[2, 2, 1]]
```

■ La función de *Mathematica* Table :

Table[exp,{i,imax}];
Table[exp,{i,imin,imax}];
Table[exp,{i,imin,imax,paso}]
Table[exp,{i,imin,imax},{j,jmin,jmax}]

Este comando permite generar listas de forma automática.
La forma más fácil Table[x,{5}],
Luego, Table[i,{i,5}], i comienza en 1 y aumenta de 1 en 1.
Si el valor inicial no es 1: Table[i,{i,2,6}]
Si el paso no es 1: Table[i,{i,1,6,2}]
Lo más complicado: una lista anidada, dos contadores. El que va en medio representa al bucle más exterior(al que varia más despacio)

```
Table[i / (i + 1), {i, 5}]
```

```
Table[i / (i + 1), {i, 5, 10}]
```

```
Table[i / (i + 1), {i, 5, 10, 2}]
```

■ Ejercicios

1- Utilice Table para hacer una lista conjunta de los cuadrados y los cubos de los números pares del uno al nueve.

2- Utilice Table para crear una lista con las potencias de x de 2 a 9 con paso 3.

3- Seleccione de la lista anterior x^5 y evalúelo para x = 2.

Matrices

En *Mathematica* una matriz es una lista de listas cada una de las cuales está integrada por los elementos de una fila.Por su parte,un vector viene dado mediante una lista (independientemente de que sea vector fila o vector columna)

Cuidado con el MatrixForm crea una especie de objeto gráfico que no se puede evaluar ni operar con él. Solo es para obtener una representación semejante a la que se hace en la pizarra.

Dot[a,b], o a.b es el producto vectorial o matricial. Aparecerán mensajes de error si se intenta multiplicar dos matrices cuyas dimensiones no concuerdan.

```
m = Table[i + j, {i, 1, 3}, {j, 2, 5}]
```

```
MatrixForm[m]
```

```
n = {{1, 2, 3}, {4, 5, 6}}
```

```
MatrixForm[n]
```

```
v = {1, 0, 0, 0}
```

```
m.v
```

- Observación

```
Clear[m]
```

```
m = Table[i + j, {i, 1, 3}, {j, 2, 5}] // MatrixForm
```

```
m.v
```

Algunos comandos de *Mathematica* de utilidad para la manipulación de matrices son:

DiagonalMatrix[{x1,x2,...,xn}]

IdentityMatrix[n]

Transpose[a]

Conjugate[a]

a[[{i1,...,ip},{j1,...,jr}]]

MatrixPower[a,n]

Det[a]

Inverse[a]
RowReduce[a]
Minors[a,k]
ZeroMatrix[m,n] (dentro del paquete <<LinearAlgebra`MatrixManipulation`)
Eigenvalues[a]
Eigenvectors[a]
Eigensystem[a]
JordanDecomposition[a]

```
mat = DiagonalMatrix[{a, b, c, d}]
```

```
IdentityMatrix[4]
```

```
Transpose[n] // MatrixForm
```

```
mat = {{1 + I, 2 - I}, {1, 3 - I}}
```

```
Conjugate[mat] // MatrixForm
```

Para sacar la submatriz formada por los elementos de las dos primeras filas y columnas 2ª y 3ª de la matriz m

```
m = Table[i + j, {i, 1, 3}, {j, 2, 5}]
```

```
MatrixForm[m]
```

```
mat = m[[{1, 2}, {2, 3}]]; MatrixForm[mat]
```

La potencia 3ª de la matriz mat (mat debe ser cuadrada)

```
MatrixPower[mat, 3]
```

El determinante y la inversa (mat debe ser cuadrada)

```
Det[mat]
```

```
Inverse[mat]
```

RowReduce realiza una versión de la eliminación gaussiana para obtener el mayor nº de ceros en la matriz resultante. Si la matriz es cuadrada y tiene inversa, el resultado es la matriz unidad. Si la última linea es de ceros, la matriz no tiene inversa, o lo que es igual, hay una combinación lineal entre sus filas.

```
RowReduce[(1 0 3
           4 2 4
           7 8 9)] // MatrixForm
```

```
RowReduce[(1 2 3
           4 5 6
           7 8 9)] // MatrixForm
```

Minors[mat,n] calcula todos los menores de la matriz mat de orden n.

```
M = {{a11, a12, a13}, {a21, a22, a23}, {a31, a32, a33}};
```

```
MatrixForm[M]
```

```
menores2 = Minors[M, 2]
```

Ejemplo de utilización de un paquete: <<LinearAlgebra`MatrixManipulation`. Dentro de este paquete hay muchos comando (ver ayuda), uno de ellos es el ZeroMatrix[n](crea una matriz cuadrada de orden n con todos los elementos iguales a 0) y ZeroMatriz[m,n] (idem. para una matriz de orden mxn). Lo primero es cargar el paquete. Para evitar errores se puede copiar de la ayuda.

```
<< LinearAlgebra`MatrixManipulation`
```

```
ZeroMatrix[3] // MatrixForm
```

El comando MatrixRank, que es nuevo en la versión 5.2, da el rango de la matriz.

```
mat = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}}; MatrixRank[mat]

2
```

Mathematica permite calcular valores y vectores propios.
Si una matriz no es diagonalizable se le aplica el comando JordanDecomposition.
Los comandos son que permiten el cálculo de autovalores y autovectores así como la obtención de la forma canónica de Jordan son: :
- Eigenvalues[m], da los valores propios de la matriz m como elementos de una lista.
- Eigenvectors[m] da una lista con los vectores propios. Cuando tiene tantos vectores propios linealmente independientes como su orden se puede diagonalizar. La matriz diagonal obtenida por transformaciones de semejanza tiene los valores propios como elementos de la diagonal principal.
- Eigensystem[m] da una lista con dos elementos, el 1º es la lista de valores propios, y el 2º es la lista de vectores propios(lista de listas). Para estos tres comando m debe ser cuadrada.
- Si una matriz m no se puede diagonalizar (no tiene tantos vectores porpios linealmente indep. como su orden) el

comando JordanDecomposition da la forma canónica de Jordan. JordanDecomposition[m] da como resultado una lista de dos elementos: {s,j}, siendo s la matriz de semejanza y j la forma canónica. Se cumple : m = s. j. s^{-1}.

```
m = (1 2 3 2
     5 3 1 8
     6 2 8 9
     4 2 1 7.); valprop = Eigenvalues[m]
```

Para Sacar uno de ellos

```
valprop[[1]]
```

```
vectprop = Eigenvectors[m]
```

```
sistemaprop = Eigensystem[m]
```

Para sacar el 2º vector propio:

```
sistemaprop[[2, 2]]
```

```
mat1 = {{2, 4, -6, 0}, {4, 6, -3, -4}, {0, 0, 4, 0}, {0, 4, -6, 2}};
Eigensystem[mat1]
```

```
{A, B} = JordanDecomposition[mat1]
```

```
MatrixForm[B]
```

Ejercicios

1- Representar una matriz 3x3 tal que $a_{ij}=x^i y^j$. Tomar un vector *p*, de dimensión 3x1, que sea la evaluación de la 1ª fila de la matriz para x=*2*,*y*=*3*.

```
```

2- Construir una matriz m, cuadrada de orden 3 con los siguientes elementos: {{a,b,c},{d,e,f},{g,h,i}}. Extraer el vector m_1 , como la 1º fila de la matriz *m*. Extraer m_2 , que sea la 2ª columna de la matriz *m*. Idem m_3 , como la diagonal principal. Idem m_4, como la otra diagonal. Obtener m_5 , una matriz 2x2 formada por los elementos esquina de *m* . Asociar también a m_6 , la matriz 2x2 adjunta al elemento (3,3).

3- Sea un vector *v1={a,b,c}*. Sea *v2={d,e,f}*. En *Mathematica* el producto elemento a elemento de vectores o matrices se realiza utilizando el simbolo **(*)**. Sin embargo el producto matricial o el escalar de vectores se consigue usando el operador **(.)**.
Multiplicar los 2 vectores elemento a elemento y escalarmente.
Repetir los cálculos con una matriz 3x2. En su caso utilizar la función Transpose[ ], cuando se necesite transponer un operando.

4- Sea A= $\begin{pmatrix} 1 & -3 & -1 & 2 \\ 2 & -4 & -1 & 3 \\ -5 & 1 & -2 & -3 \\ 3 & 5 & 4 & -1 \end{pmatrix}$.

a)Hallar el valor del determinante de A.
b) Hallar el rango de A.

5- Hallar la inversa de la matriz A= $\begin{pmatrix} 2 & 1 & 1 \\ 4 & 2 & 0 \\ -3 & -1 & 1 \end{pmatrix}$.

6- Calcular los valores y vectores propios de las matrices:

A= $\begin{pmatrix} 1 & 2 & 2 \\ 2 & 1 & 2 \\ 2 & 2 & 1 \end{pmatrix}$ B= $\begin{pmatrix} m & 0 & 2-m \\ 0 & 4 & 0 \\ -m & 0 & 2+m \end{pmatrix}$

7- Dada la matriz A= $\begin{pmatrix} 1 & 0 & 1 & 0 \\ -1 & -1 & 4 & 0 \\ -1 & 0 & 3 & 0 \\ -1 & 0 & 4 & -1 \end{pmatrix}$ hallar la matriz J que sea su forma canónica de Jordan, así como la matriz P que relaciona ambas matrices