

MATHEMATICA

Actividades de Clase: 40%

Tareas: 30%

Examén: 20%

Asistencia: 10%

¿Qué es Wolfram Language?

Wolfram Language es un *lenguaje de computación*. Es una forma de comunicarse con una computadora; en particular, para decirle lo que uno quiere que haga.

Programa

- Aritmética Elemental
- Datos del mundo Real
- Introducción a las funciones
- Un primer vistazo a las listas
- Visualización de listas
- Operaciones con listas
- Colores y estilos

Cálculo Diferencial e Integral

Gráficas 2D y 3D

Contornos

Condicionales

Conjuntos de Datos

Importar y Exportar

Escribir un código

ARITMÉTICA ELEMENTAL

2+2 adición

5-2 sustracción

2*3 multiplicación (2 3 también funciona)

6/2 división

3^2 elevación a una potencia (por ejemplo, al cuadrado)

- Calcular $1+2+3$.
- Sumar los números 1, 2, 3, 4, 5.
- Multiplicar los números 1, 2, 3, 4, 5.
- Calcular 5 al cuadrado
- Calcular 3 elevado a la cuarta potencia.
- Calcular 10 elevado a la potencia 12.
- Calcular 3 elevado a la potencia 7×8 .
- Colocar los paréntesis necesarios para que $4-2*3+4$ sea igual a 14.
- Calcular veintinueve mil multiplicado por setenta y tres.
- Sumar los enteros entre -3 y +3.

ARITMÉTICA ELEMENTAL

Vocabulario

Plus[2,2]	2+2	adición
Subtract[5,2]	5-2	sustracción
Times[2,3]	2*3	multiplicación (también puede escribirse como 2 3)
Divide[6,2]	6/2	división
Power[3,2]	3^2	eleva a una potencia
Max[3,4]		máximo (el mayor)
Min[3,4]		mínimo (el menor)
RandomInteger[10]		número entero aleatorio

- Calcule $7+6+5$ usando la función Plus.
- Calcule $2 \times (3+4)$ usando Times y Plus.
- Utilice Max para encontrar el máximo entre 6×8 y 5×9 .
- Use RandomInteger para generar un número aleatorio entre 0 y 1000.
- Use Plus y RandomInteger para generar un número entre 10 y 20.
- Calcule $5 \times 4 \times 3 \times 2$ usando Times.
- Calcule $2-3$ usando Subtract.
- Calcule $(8+7) \times (9+2)$ usando Times y Plus.
- Calcule $(26-89)/9$ usando Subtract y Divide.
- Calcule $100-5^2$ usando Subtract y Power.
- Encuentre el mayor entre 3^5 y 5^3 .
- Multiplique 3 por el mayor entre 4^3 y 3^4 .

Vocabulario

UnitConvert[cantidad, unidad] convierte entre unidades

CurrencyConvert[monto, unidad] convierte entre divisas

Convierta 4.5 lbs a kilogramos.

Convierta 60.25 mph a kilómetros por hora.

Convierta 2500 yenes japoneses a dólares US. »

Encuentre el total de 35 onzas, más 1/4 ton (ton = 907 kg), más 45 lbs, más 9 stone (stone = 14 lbs), en kilogramos.

Divisa	Abreviatura	Símbolo
Dólar estadounidense	USD	US\$
Euro	EUR	€
Yen japonés	JPY	¥
Libra esterlina	GBP	£
Dólar australiano	AUD	A\$
Dólar canadiense	CAD	C\$
Franco suizo	CHF	CHF
Renminbi/yuan chino	CNY	元
Dólar de Hong Kong	HKD	HK\$

Dólar de Nueva Zelanda	NZD	NZ\$
Corona sueca	SEK	kr
Won surcoreano	KRW	₩
Dólar de Singapur	SGD	S\$
Corona noruega	NOK	kr
Peso mexicano	MXN	\$
Rupia india	INR	₹
Rublo ruso	RUB	₽
Rand sudafricano	ZAR	R
Lira turca	TRY	₺

LISTAS

Vocabulario

{1,2,3,4}	lista de elementos
ListPlot[{1,2,3,4}]	presenta gráficamente una lista de números
Range[10]	una secuencia de números consecutivos
Reverse[{1,2,3}]	invierte el orden de una lista
Join[{4,5,6},{2,3,2}]	junta varias listas

- Use Range para crear la lista {1, 2, 3, 4}.
- Construya la lista de los números hasta el 100.
- Use Range y Reverse para crear {4, 3, 2, 1}.
- Construya la lista de los números del 1 al 50 en orden inverso.
- Use Range, Reverse y Join para crear {1, 2, 3, 4, 4, 3, 2, 1}.
- Use Range y RandomInteger para crear una lista de longitud aleatoria hasta 10.
- Encuentre una forma más simple para Reverse[Reverse[Range[10]]].
- Encuentre una forma más simple para Join[{1, 2}, Join[{3, 4}, {5}]].
- Encuentre una forma más simple para Join[Range[10], Join[Range[10], Range[5]]].

LISTAS

- Construya un diagrama de barras con {1, 1, 2, 3, 5}.
- Produzca un diagrama circular con los números del 1 al 10.
- Forme un diagrama de barras de los números consecutivos del 20 al 1.
- Muestre en una columna los números del 1 al 5.
- Presente los cuadrados {1, 4, 9, 16, 25} sobre una recta numérica.
- Forme un gráfico circular con 10 sectores idénticos, cada uno de tamaño 1.
- Presente una columna de los gráficos circulares con 1, 2 y 3 sectores idénticos.
- Forme una lista de gráficos circulares con 1, 2 y 3 sectores idénticos.
- Presente un diagrama de barras con la secuencia 1, 2, 3, ..., 9, 10, 9, 8, 7, ..., 1.

ListLinePlot[{1,2,5}]	valores unidos por segmentos de recta
BarChart[{1,2,5}]	diagrama de barras (los valores dan la altura de las barras)
PieChart[{1,2,5}]	diagrama circular (los valores dan el tamaño de los sectores)
NumberLinePlot[{1,2,5}]	números colocados sobre una recta
Column[{1,2,5}]	elementos exhibidos en una columna

LISTAS

Vocabulario

<code>{2,3,4}+{5,6,2}</code>	aritmética con listas
<code>Sort[{5,7,1}]</code>	ordena una lista
<code>Length[{3,3}]</code>	longitud de una lista (número de elementos)
<code>Total[{1,1,2}]</code>	total de los elementos de una lista
<code>Count[{3,2,3},3]</code>	cuenta cuántas veces aparece un elemento
<code>First[{2,3}]</code>	primer elemento de una lista
<code>Last[{6,7,8}]</code>	último elemento de una lista
<code>Part[{3,1,4},2]</code>	un elemento particular de una lista, que también se escribe como <code>{3,1,4}[[2]]</code>
<code>Take[{6,4,3,1},2]</code>	toma elementos del principio de la lista
<code>Drop[{6,4,3,1},2]</code>	desecha elementos del principio de una lista
<code>IntegerDigits[1234]</code>	lista de los dígitos de un número

- Cree una lista de los 10 primeros cuadrados en orden inverso.
- Calcule el total de los 10 primeros cuadrados.
- Muestre gráficamente los 10 primeros cuadrados, comenzando por el 1.
- Cree una lista de los 10 primeros múltiplos de 3.
- Use Sort, Join y Range para crear `{1, 1, 2, 2, 3, 3, 4, 4}`.
- Use Range y + para formar la lista de los números del 10 al 20, inclusive.
- Forme la lista de los 10 primeros cuadrados, usando únicamente Range y Times.
- Encuentre el número de dígitos en 2^{128} .
- Encuentre el primer dígito de 2^{32} .
- Encuentre los 10 primeros dígitos en 2^{100} .
- Encuentre el último dígito de 2^{37} .
- Encuentre el penúltimo dígito de 2^{32} .

LISTAS

Vocabulario

Table[x,5]	lista de 5 copias de x
Table[f[n],{n,10}]	lista de valores de f[n] donde n varía hasta el 10
Table[f[n],{n,2,10}]	lista de valores donde n varía desde el 2 hasta el 10
Table[f[n],{n,2,10,4}]	lista de valores donde n varía del 2 hasta el 10, de 4 en 4
Range[5,10]	lista de los números del 5 al 10
Range[10,20,2]	lista de los números del 10 al 20, de 2 en 2
RandomInteger[10,20]	lista de 20 enteros aleatorios entre 0 y 10

- Forme una lista con el número 1000 repetido 5 veces.
- Cree una lista de los valores de n^3 donde n va del 10 al 20.
- Construya un gráfico con los 20 primeros cuadrados sobre la recta numérica.
- Forme una lista de las diferencias entre n^3 y n^2 para n menor o igual a 10.
- Construya una lista de los números pares (2, 4, 6, ...) hasta el 20.
- Forme una lista de los números impares (1, 3, 5, ...) hasta el 100.
- Cree una lista de los cuadrados de los números pares hasta el 100.
- Cree la lista {-3, -2, -1, 0, 1, 2} utilizando Range.
- Use Table para obtener el mismo resultado que Range[10].
- Construya un diagrama de barras de los 10 primeros cuadrados.
- Forme una tabla de listas de los dígitos de los 10 primeros cuadrados.
- Forme una tabla del primer dígito de cada uno de los 20 primeros cuadrados.

COLORES

Vocabulario

Red, Green, Blue, Yellow, Orange, Pink, Purple, ...	colores
RGBColor[0.4,0.7,0.3]	color rojo, verde, azul
Hue[0.8]	color especificado mediante tonalidad
RandomColor[]	color escogido al azar
ColorNegate[Red]	negativo de un color (complemento)
Blend[{Red,Blue}]	mezcla una lista de colores
Style[x,Red]	estilo con un color
Style[x,20]	estilo con un tamaño
Style[x,20,Red]	estilo con un tamaño y un color

Forme una lista de rojo, amarillo y verde.

Producza una columna de rojo, amarillo y verde ("un semáforo").

Calcule el negativo del color naranja.

Construya una lista con tonalidades que varíen del 0 al 1 en incrementos de 0.02.

Mezcle los colores rosa y amarillo.

Mezcle el color rojo con su negativo.

Obtenga la mezcla de colores con tonalidades del 0 al 1 en incrementos de 0.1.

Mezcle el color rojo con el blanco y mezcle el resultado de nuevo con el blanco.

Formar una lista de 100 colores aleatorios.

Muestre un retal en color morado de tamaño 100.

Forme una lista de retales rojos de tamaños del 10 al 100 en incrementos de 10.

Vocabulario

Circle[]	especifica un círculo
Disk[]	especifica un disco relleno
RegularPolygon[<i>n</i>]	especifica un polígono regular con <i>n</i> lados
Graphics[<i>objeto</i>]	muestra un objeto gráficamente
Sphere[], Cylinder[], Cone[], ...	especifica formas geométricas en 3D
Graphics3D[<i>objeto</i>]	muestra un objeto gráficamente en 3D

Use RegularPolygon para dibujar un triángulo. [»](#)

.1

Dibuje el gráfico de un círculo rojo. [»](#)

1.1

Muestre un octágono rojo. [»](#)

1.1

Forme una lista cuyos elementos sean discos en tonalidades que varíen de 0 a 1 en incrementos de 0.1. [»](#)

1.1

Muestre una columna con un triángulo rojo y otro verde. [»](#)

1.1

Presente una lista con los polígonos regulares que tengan de 5 a 10 lados, con cada polígono en color rosa. [»](#)

1.1

Forme una lista de 10 pentágonos regulares coloreados de manera aleatoria. [»](#)

1.1

Construya una lista con un polígono regular de 20 lados y un disco. [»](#)

1.1

Cree una lista de polígonos con 10, 9, ..., 3 lados. [»](#)

1.1

Muestre el gráfico de un cilindro morado. [»](#)

1.1

Forme una lista de polígonos con 8, 7, 6, ..., 3 lados, coloreados aleatoriamente con RandomColor, y muéstrela sobrepujando con el triángulo arriba (sugerencia: aplicar Graphics a la lista). [»](#)

1.1

Vocabulario

Manipulate[*algo*,{*n*,0,10,1}] manipula *algo* donde *n* varía de 1 en 1

Manipulate[*algo*,{*x*,0,10}] manipula *algo* donde *x* varía de manera continua

- Cree un Manipulate para graficar números del 1 al 100, cada uno elevado a alguna potencia entre -1 y $+1$.
- Formule un Manipulate que muestre secuencias de los dígitos de enteros de 4 dígitos (entre 1000 y 9999).
- Escriba un Manipulate para crear una lista de entre 5 y 50 tonalidades igualmente espaciadas.
- Cree un Manipulate para mostrar una lista de un número variable de hexágonos (entre 1 y 10), con tonalidades variables.
- Escriba un Manipulate que muestre un polígono regular con entre 5 y 20 lados, de colores rojo, amarillo o azul.
- Cree un Manipulate para mostrar los valores de x^n en una recta numérica , donde *x* sea un entero entre 1 y 10, y donde *n* varíe entre 0 y 5.
- Escriba un Manipulate que muestre una esfera tal que su color varíe entre el verde y el rojo.

Vocabulario

$a == b$	comprueba igualdad
$a < b$	comprueba si es menor que
$a > b$	comprueba si es mayor que
$a \leq b$	comprueba si es menor o igual
$a \geq b$	comprueba si es mayor o igual
If [<i>test</i> , <i>u</i> , <i>v</i>]	obtiene el resultado <i>u</i> si <i>test</i> es True y <i>v</i> si es False
Select [<i>lista</i> , <i>f</i>]	selecciona los elementos que satisfacen un criterio
EvenQ [<i>x</i>]	comprueba si es par
OddQ [<i>x</i>]	comprueba si es impar
IntegerQ [<i>x</i>]	comprueba si es un entero
PrimeQ [<i>x</i>]	comprueba si es un número primo
LetterQ [<i>cadena</i>]	comprueba si solo son letras
MemberQ [<i>lista</i> , <i>x</i>]	comprueba si <i>x</i> está contenido en <i>lista</i>
ImageInstanceQ [<i>imagen</i> , <i>categoria</i>]	comprueba si <i>imagen</i> es un caso de <i>categoria</i>

Cálculo Diferencial e Integral

Límites

Limit $[f, x \rightarrow x^*]$

gives the limit $\lim_{x \rightarrow x^*} f(x)$.

Limit $[f, \{x_1 \rightarrow x_1^*, \dots, x_n \rightarrow x_n^*\}]$

gives the nested limit $\lim_{x_1 \rightarrow x_1^*} \dots \lim_{x_n \rightarrow x_n^*} f(x_1, \dots, x_n)$.

Limit $[f, \{x_1, \dots, x_n\} \rightarrow \{x_1^*, \dots, x_n^*\}]$

gives the multivariate limit $\lim_{\{x_1, \dots, x_n\} \rightarrow \{x_1^*, \dots, x_n^*\}} f(x_1, \dots, x_n)$.

Cálculo Diferencial e Integral

Derivadas

D (∂)

D [f, x]

gives the partial derivative $\partial f / \partial x$.

D [$f, \{x, n\}$]

gives the multiple derivative $\partial^n f / \partial x^n$.

D [f, x, y, \dots]

gives the partial derivative $\dots (\partial / \partial y) (\partial / \partial x)f$.

D [$f, \{x, n\}, \{y, m\}, \dots$]

gives the multiple partial derivative $\dots (\partial^m / \partial y^m) (\partial^n / \partial x^n)f$.

D [$f, \{\{x_1, x_2, \dots\}\}$]

for a scalar f gives the vector derivative $(\partial f / \partial x_1, \partial f / \partial x_2, \dots)$.

D [$f, \{array\}$]

gives an array derivative.

ListPlot

UPDATED IN
13

`ListPlot[{y1, ..., yn}]`

plots regularly spaced points {1, y₁}, {2, y₂},

`ListPlot[{{x1, y1}, ..., {xn, yn}]}`

generates a scatter plot with points {x_i, y_i}.

`ListPlot[{data1, data2, ...}]`

plots points from all the data_i.

`ListPlot[{..., w[datai, ...], ...}]`

plots data_i with features defined by the symbolic wrapper w.

`Show[graphics, options]`

shows graphics with the specified options added.

`Show[g1, g2, ...]`

shows several graphics combined.

Fit

Fit [*data*, $\{f_1, \dots, f_n\}$, $\{x, y, \dots\}$]

finds a fit $a_1 f_1 + \dots + a_n f_n$ to a list of *data* for functions f_1, \dots, f_n of variables $\{x, y, \dots\}$.

Fit [$\{m, v\}$]

finds a fit vector a that minimizes $\|m.a - v\|$ for a design matrix m .

Fit [..., "prop"]

specifies what fit property *prop* should be returned.

FindFit

FindFit [*data*, *expr*, *pars*, *vars*]

finds numerical values of the parameters *pars* that make *expr* give a best fit to *data* as a function of *vars*.

FindFit [*data*, $\{expr, cons\}$, *pars*, *vars*]

finds a best fit subject to the parameter constraints *cons*.

Export

Export ["*dest.ext*", *expr*]

exports data to a file, converting it to the format corresponding to the file extension *ext*.

Export [*dest*, *expr*, "*fmt*"]

exports data in the specified format "*fmt*".

Export [*dest*, *exprs*, *elements*]

exports data by treating *exprs* as *elements*.

Export [*dest*, *exprs*, *elements*, *options*]

uses the specified options.

Import

Import [*source*]

imports data from *source*, returning a Wolfram Language representation of it.

Import [*source*, "*fmt*"]

takes the file to be in the specified format "*fmt*".

Import [*source*, *elements*]

imports the specified *elements* from a file.

Import [*source*, ..., *options*]

uses the specified options.