

Tutoría 4

Física Computacional I

Grado en Física



UNED

Javier Carrasco Serrano, javcarrasco@madrid.uned.es

Física Computacional I, Las Tablas

Repaso funciones

- Única expresión:

“nombre de la función” (lista de argumentos separados por comas) := expresión que define la función ;

```
(%i1) f(x) := x^2;      (%i1) f(a);
```

- Varias expresiones:

“nombre de la función” (lista de argumentos separados por comas) := block ([lista de variables locales separadas por comas] , lista de expresiones necesarias para definir la función separadas por comas, return(expresión retornada por la función));

Bucles y condicionales

- Pseudocódigo: <https://es.wikipedia.org/wiki/Pseudoc%C3%B3digo>
- Bucle: repetir una o varias sentencias un número determinado (o no) de veces.
 1. Para $i=\text{valor_inicial}$ hasta $i=\text{valor_final}$ ejecutar {sentencias}
 2. Para cada i en los elementos de una lista ejecutar {sentencias}
 3. Para $i=\text{valor_inicial}$ hasta $i=\text{valor_final}$ ejecutar {sentencias} mientras se cumpla condición
 4. Para $i=\text{valor_inicial}$ hasta $i=\text{valor_final}$ ejecutar {sentencias} hasta que se cumpla condición
- Sentencia condicional: ejecutar una o varias sentencias dependiendo si se cumple o no una condición.
 1. Si {condición lógica} entonces {sentencias}
 2. Si {condición lógica} entonces {sentencias}, si no {sentencias}
- Combinaciones de sentencias condicionales y bucles.

Bucles y condicionales

- Bucles Maxima: http://maxima.sourceforge.net/docs/manual/es/maxima_37.html
- Bucle: repetir una o varias sentencias un número determinado (o no) de veces.
 1. *for variable: valor_inicial step incremento thru limite do cuerpo*
 1. *for variable from valor_inicial step incremento thru limite do cuerpo*
 2. *for variable in lista do cuerpo*
 3. *for variable: valor_inicial step incremento while condicion do cuerpo*
 4. *for variable: valor_inicial step incremento unless condición do cuerpo*

Atención: la variable de control es una variable local.

“condicion” es una expresión lógica (booleano) que se evalúa como verdadera (TRUE) o falsa (FALSE).

Si queremos que se ejecuten varias sentencias, tenemos que ponerlas entre paréntesis y separadas por comas: (sentencia1, sentencia2, sentencia3)

Bucles y condicionales

- Sentencias condicionales Maxima: http://maxima.sourceforge.net/docs/manual/es/maxima_37.html
- Sentencia condicional: ejecutar una o varias sentencias dependiendo si se cumple o no una condición.
 1. If condicion then sentencia1
 2. If condicion then sentencia1 else sentencia2
 3. Los bucles condicionales se pueden anidar si en la sentencia2 del else escribimos otra sentencia if-else, o si en su lugar utilizamos el comando elseif: `if cond_1 then expr_1 elseif cond_2 then expr_2`

Atención: la variable de control es una variable global.

“condicion” es una expresión lógica (booleano) que se evalúa como verdadera (TRUE) o falsa (FALSE).

Si queremos que se ejecuten varias sentencias, tenemos que ponerlas entre paréntesis y separadas por comas: (sentencia1, sentencia2, sentencia3)

Bucles y condicionales

- Algunos operadores lógicos: http://maxima.sourceforge.net/docs/manual/es/maxima_6.html#SEC43

Operación	Símbolo	Tipo
menor que	<	operador relacional infijo
menor o igual que	<=	operador relacional infijo
igualdad (sintáctica)	=	operador relacional infijo
negación de =	#	operador relacional infijo
igualdad (por valor)	equal	operador relacional infijo
negación de equal	notequal	operador relacional infijo
mayor o igual que	>=	operador relacional infijo
mayor que	>	operador relacional infijo
y	and	operador lógico infijo
o	or	operador lógico infijo
no	not	operador lógico prefijo

Bucles y condicionales

- Tipos de datos: https://es.wikipedia.org/wiki/Tipo_de_dato
- Tipos de datos en Maxima: http://maxima.sourceforge.net/docs/manual/es/maxima_5.html#SEC18
- El tipo de datos booleano es muy importante por su uso en bucles y condicionales. Son datos que sólo toman dos valores, TRUE o FALSE, dependiendo de si la expresión lógica es verdadera o falsa.

2.6. Aprendiendo Maxima

- Guardar archivos desde sesiones de trabajo: archivo → exportar → fichero por lotes maxima (*.mac).
- Es más, para trabajar es mucho mejor guardar en ficheros de texto plano funciones etc, y cargarlas en máxima para su uso.
- Guarrería: copiar y pegar.
- Menos guarrería: archivo → fichero de lotes.
- Recomendable para cargar ficheros → `batchload (filename);`

Filename es el nombre con directorio del fichero:

C:\Users\Documents\Javier\uned\Tutor\Fisica_Computadores_\Tutorías\Tutoría 2 - 19 Febrero\ejemplo.mac

Por defecto busca sólo en los directorios guardados en la variable `file_search_máxima`.

Se pueden añadir → `append(file_search_máxima, ["mi_directorio"]);`

En la práctica es más cómodo asignar el directorio a una variable global, y llamarla cada vez que hace falta:

```
path_mec : "/home/usuario/bib/Maxima/mecanica/";  
(obsérvese el uso de comillas dobles, a fin de que la variable path_mec sea de tipo string) y posteriormente empleamos la función de concatenación concat para generar el path completo de los archivos a cargar
```

```
batchload( concat( path_mec, "ec-Newton.mc" ) );  
batchload( concat( path_mec, "ecs-Euler-Lagrange.mc" ) );
```


Tema 6:
Solución analítica
y numérica de
ecuaciones

06

Algunos comandos básicas

Maxima permite resolver ecuaciones y sistemas de ecuaciones. Algunos comandos para ello son:

- Ecuaciones algebraicas: *solve (ecuacion, variable);*
- Sistema de ecuaciones algebraicas: *solve (lista_de_ecuaciones, lista_de_variables);*
- Aproximación numérica de ecuaciones – método de Newton: *newton (ecuación, variable, valor_inicial, precisión);*
 - se tiene que cargar primero la librería!!! → *load("newton1");*

https://es.wikipedia.org/wiki/M%C3%A9todo_de_Newton

6.4. Funciones de Maxima que debemos utilizar

- *Sublis* →
- *Length* → calcula la cantidad de elementos de una lista.
- *Apply* → permite suministrar a otra función todos los elementos de una lista como argumentos, así podemos sumar o multiplicar todos los elementos de la lista directamente, aunque también convierte una lista de datos en una matriz (lo veremos en el tema siguiente)
- *Subst, diff, block, append, makelist, plot2d, plot3d...*

6.1. Función implícita

- Función implícita: https://es.wikipedia.org/wiki/Funci%C3%B3n_impl%C3%ADcita
- Teorema de la función implícita:
https://es.wikipedia.org/wiki/Teorema_de_la_funci%C3%B3n_impl%C3%ADcita
- Aproximaciones con el método de newton o polinomios de Taylor.

6.3. Función inversa

- Teorema de la función inversa: https://es.wikipedia.org/wiki/Teorema_de_la_funci%C3%B3n_inversa

6.5. Problemas resueltos



Adobe Acrobat
Document

Problemas del Tema 6

Algunos ejercicios resueltos de probabilidad:



Adobe Acrobat
Document



Adobe Acrobat
Document

Gracias!



UNED