= Breve discusión de métodos numéricos =

Tipos de lenguajes de programación:

Bajo Nivel	Alto Nivel
Muy parecidos a como "habla la computadora"	Más parecidos al lenguaje humano
• C, Fortran	• Python, Julia, Matlab

Paradigmas de la programación

Procedimiento: Como C donde todos los datos se tratan de forma iterativa con funciones

C, C++, Fortran, Python

Funcional: Funciones y datos se operan al mismo nivel y se dan instrucciones de su uso para al final sólo evaluarse una vez.

Mathematica, Lisp, Haskel, R

Mathematica está escrito en C++ (Procedimiento) pero es un lenguaje funcional.

↳ También puede programarse en procedimiento pero es menos eficiente

¿Cómo es distinta la programación según el paradigma?

Procedimiento:

```
MovingAverage1[z_, k_] :=
Module[{zav = {}, i, j, temp},
    Do[ temp = 0;
        Do[temp += z[[i + j]], {j, 0, k}];
        AppendTo[zav, temp/(k + 1)],
        {i, 1, Length[z] - k}
    ];
    zav
]
```

5 líneas

```
nums = Range[1., 1000., 1.];

Timing[ MovingAverage1[nums, 100]; ]
{34.6333 Second, Null}
```

Se tarda 34s

Funcional:

```
MovingAverage2[z_, k_] :=
   Plus @@ Partition[z, Length[z] - k, 1]/(k + 1)
```

← 1 línea

```
Timing[ MovingAverage2[nums, 100]; ]
{2.56667 Second, Null}
```

Se tarda 2s.

↳ Sin embargo, C o Fortran lo harían más rápido por ser de bajo nivel.

¿Cómo se resuelven ecuaciones diferenciales?

Si $\frac{d}{dt}y = f(y,t) \Rightarrow \overset{\text{primera aprox}}{\frac{\Delta y}{\Delta t}} = f(y,t) \Rightarrow \Delta y = y_i - y_{i-1} = f(y_{i-1}, t)\Delta t$

$\Rightarrow y_i = f(y_{i-1},t)\Delta t + y_{i-1}$ → y eso iterativamente

↳ Necesito 1 condición inicial

* Hay más métodos pero este es el más simple.

Si tengo ecuaciones de segundo orden, me conviene convertirlas en dos de segundo orden y resolverlas simultáneamente

$$\frac{d^2}{dt^2}y = f\left(\frac{dy}{dt}, y, t\right) \longrightarrow \begin{cases} \frac{dy}{dt} = V_y & \text{→ Necesito 1 condición inicial} \\ \frac{d^2}{dt^2}y = \frac{d}{dt}V_y = f(V_y, y, t) & \text{→ Necesito 1 condición inicial} \end{cases}$$

Necesitamos 2 condiciones iniciales.

↓ Necesitamos 2 condiciones iniciales

El sistema $\begin{cases} \frac{dy}{dt} = V_y \\ \frac{d}{dt}V_y = f(V_y, y, t) \end{cases}$ lo pongo en un método numérico y ya quedó

Hagamos el ejemplo con caída libre

$$T = \frac{1}{2} m \dot{y}^2, \quad V = mg(l - y) \Rightarrow \mathcal{L} = \frac{1}{2} m \dot{y}^2 - mg(l-y)$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial}{\partial \dot{y}} \mathcal{L}\right) - \frac{\partial}{\partial y}\mathcal{L} = m\ddot{y} + mg = 0 \Rightarrow \ddot{y} = -g \longrightarrow$$

Definimos

$$\dot{y} = v_y$$

$$\dot{v}_y = -g$$

Script para resolver las ecuaciones diferenciales

```
toMap = If[Length[#] == 0, {#}, #] &
```

```
If[Length[#1] == 0, {#1}, #1] &
```

```
solver[eqs_, func_, var_, iCondtions_, {t0_, tf_, dt_}] := Module[{time, sol},
  time = Range[t0, tf, dt];
  sol = NDSolve[{eqs, iCondtions}, toMap[func], {var, t0, tf}];
  sol = Map[#[var] /. sol[[1]] /. var → time &, toMap[func]];
  Join[{time}, sol]]
```

```
eqs = {x'[t] == dx[t],
       dx'[t] == - 9.81};
ci = {x[0] == 0, dx[0] == 50};
{time, y, vy} = solver[eqs, {x, dx}, t, ci, {0, 15, .1}];
```

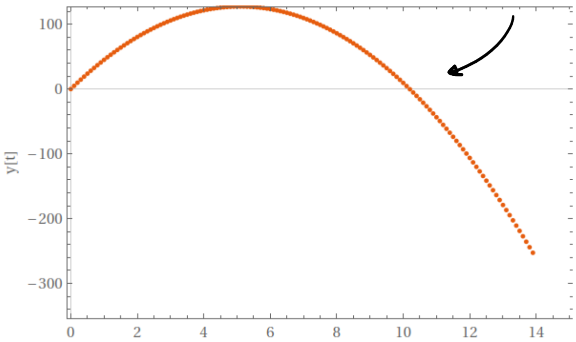
$\longrightarrow \dot{x} = v_x$

$\dot{v}_x = -g$

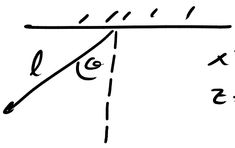
$\longrightarrow x(0) = 0, \ \dot{x}(0) = 50$

$\underbrace{\{time, y, vy\}}_{\text{soluciones}}$

Resultados obtenidos para $y(t)$



Hagamos el ejemplo del péndulo



$$x^2 + y^2 - l^2 = 0 \qquad \{q\} \to \{\theta\}$$

$$z = 0$$

$$\mathcal{L} = \underbrace{\frac{1}{2} m l^2 \dot{\theta}^2}_{T} - \underbrace{mgl(1 - \cos\theta)}_{V}$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial}{\partial \dot{\theta}}\mathcal{L}\right) - \frac{\partial \mathcal{L}}{\partial \theta} = ml^2\ddot{\theta} + mgl\sin\theta = 0$$

$$\Rightarrow \ddot{\theta} = -\frac{g}{l}\sin\theta \quad \text{sin solución analítica}$$

$$\Rightarrow \ddot{\theta} \simeq -\frac{g}{l}\theta \quad \text{aproximación de ángulo chico}$$

$\theta \ll 1$

$$\theta = A\sin\sqrt{\frac{g}{l}}\,t + B\cos\sqrt{\frac{g}{l}}\,t$$

$t_0 \quad t_f \quad \Delta t$

```
time = {0, 2*Pi, .01};
g = 9.81;
ell = 5;

eqsReal = {th'[t] == dth[t],
           dth'[t] == - (g / ell) * Sin[th[t]]};
eqsSimple = {th'[t] == dth[t],
           dth'[t] == - (g / ell) * th[t]};
ci = {th[0] == -Pi / 4, dth[0] == 0};
{tt, θ1, dθ1} = solver[eqsReal, {th, dth}, t, ci, time];
{tt, θ2, dθ2} = solver[eqsSimple, {th, dth}, t, ci, time];
```

$\longrightarrow g$

$\longrightarrow l$

$\dot{\theta} = v_\theta$

$\dot{v}_\theta = -\frac{g}{l}\sin(\theta)$

$\dot{\theta} = v_\theta$

$\dot{v}_\theta = -\frac{g}{l}\theta$

y los resultados obtenidos son:

para $\dot{\theta}(0) = 0$ y $\theta(0) = -\pi/4 = -45°$