



Universidad Nacional
de La Matanza



Curso de ingeniería centrado en código

Capitalizando lo desarrollado durante el confinamiento

Bettachini, V.A.

Ingeniería Mecánica, DIIT, UNLaM

V Encuentro *Mejora de las Estrategias Pedagógicas*
22 de septiembre de 2023



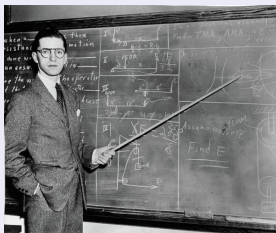
Docentes y alumnos pueden utilizar mejor su tiempo

Licklider (1957): 85 % de “pensar” es lo mundano (calcular, dibujar, etc.)



Docentes y alumnos pueden utilizar mejor su tiempo

Licklider (1957): 85 % de “pensar” es lo mundano (calcular, dibujar, etc.)



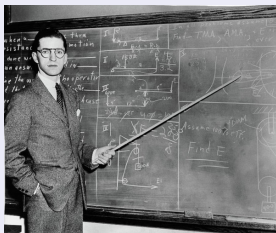
Aula y práctica: transcripción y reiteración

- Memoria $\xrightarrow{\text{profesor}}$ pizarrón/presentación



Docentes y alumnos pueden utilizar mejor su tiempo

Licklider (1957): 85 % de “pensar” es lo mundano (calcular, dibujar, etc.)



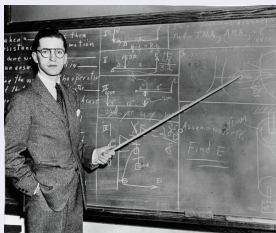
Aula y práctica: transcripción y reiteración

- Memoria $\xrightarrow{\text{profesor}}$ pizarrón/presentación
- Pizarrón/presentación $\xrightarrow{\text{alumno}}$ cuaderno



Docentes y alumnos pueden utilizar mejor su tiempo

Licklider (1957): 85 % de “pensar” es lo mundano (calcular, dibujar, etc.)



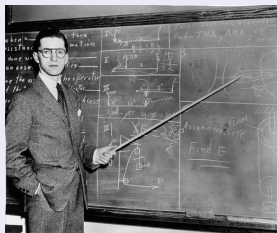
Aula y práctica: transcripción y reiteración

- Memoria $\xrightarrow{\text{profesor}}$ pizarrón/presentación
- Pizarrón/presentación $\xrightarrow{\text{alumno}}$ cuaderno
- Práctica: **reiterar** diagramas, cálculos, etc.



Docentes y alumnos pueden utilizar mejor su tiempo

Licklider (1957): 85 % de “pensar” es lo mundano (calcular, dibujar, etc.)



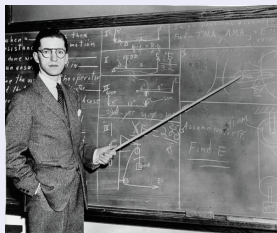
Aula y práctica: transcripción y reiteración

- Memoria $\xrightarrow{\text{profesor}}$ pizarrón/presentación
- Pizarrón/presentación $\xrightarrow{\text{alumno}}$ cuaderno
- Práctica: **reiterar** diagramas, cálculos, etc.
- Aburrimiento $\implies \downarrow$ concentración



Docentes y alumnos pueden utilizar mejor su tiempo

Licklider (1957): 85 % de “pensar” es lo mundano (calcular, dibujar, etc.)



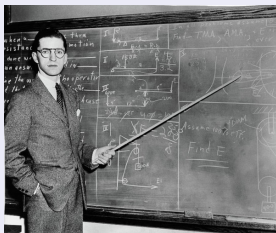
Aula y práctica: transcripción y reiteración

- Memoria $\xrightarrow{\text{profesor}}$ pizarrón/presentación
- Pizarrón/presentación $\xrightarrow{\text{alumno}}$ cuaderno
- Práctica: **reiterar** diagramas, cálculos, etc.
- Aburrimiento $\implies \downarrow$ concentración



Docentes y alumnos pueden utilizar mejor su tiempo

Licklider (1957): 85 % de “pensar” es lo mundano (calcular, dibujar, etc.)



Aula y práctica: transcripción y reiteración

- Memoria $\xrightarrow{\text{profesor}}$ pizarrón/presentación
- Pizarrón/presentación $\xrightarrow{\text{alumno}}$ cuaderno
- Práctica: **reiterar** diagramas, cálculos, etc.
- Aburrimiento $\implies \downarrow$ concentración

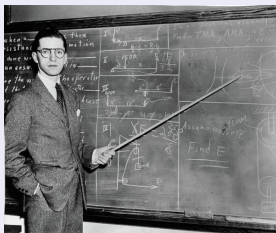
```
1 // Problema 1: Calcular la suma de los cuadrados de los números de 1 a n.
2 // El usuario ingresa un número n, y el programa calcula la suma de los cuadrados de los números de 1 a n.
3 // El programa muestra el resultado en la consola.
4 // El programa muestra el resultado en la consola.
5 // El programa muestra el resultado en la consola.
6 // El programa muestra el resultado en la consola.
7 // El programa muestra el resultado en la consola.
8 // El programa muestra el resultado en la consola.
9 // El programa muestra el resultado en la consola.
10 // El programa muestra el resultado en la consola.
11 // El programa muestra el resultado en la consola.
12 // El programa muestra el resultado en la consola.
13 // El programa muestra el resultado en la consola.
14 // El programa muestra el resultado en la consola.
15 // El programa muestra el resultado en la consola.
16 // El programa muestra el resultado en la consola.
17 // El programa muestra el resultado en la consola.
18 // El programa muestra el resultado en la consola.
19 // El programa muestra el resultado en la consola.
20 // El programa muestra el resultado en la consola.
21 // El programa muestra el resultado en la consola.
22 // El programa muestra el resultado en la consola.
23 // El programa muestra el resultado en la consola.
24 // El programa muestra el resultado en la consola.
25 // El programa muestra el resultado en la consola.
26 // El programa muestra el resultado en la consola.
27 // El programa muestra el resultado en la consola.
28 // El programa muestra el resultado en la consola.
29 // El programa muestra el resultado en la consola.
30 // El programa muestra el resultado en la consola.
31 // El programa muestra el resultado en la consola.
32 // El programa muestra el resultado en la consola.
33 // El programa muestra el resultado en la consola.
34 // El programa muestra el resultado en la consola.
35 // El programa muestra el resultado en la consola.
36 // El programa muestra el resultado en la consola.
37 // El programa muestra el resultado en la consola.
38 // El programa muestra el resultado en la consola.
39 // El programa muestra el resultado en la consola.
40 // El programa muestra el resultado en la consola.
41 // El programa muestra el resultado en la consola.
42 // El programa muestra el resultado en la consola.
43 // El programa muestra el resultado en la consola.
44 // El programa muestra el resultado en la consola.
45 // El programa muestra el resultado en la consola.
46 // El programa muestra el resultado en la consola.
47 // El programa muestra el resultado en la consola.
48 // El programa muestra el resultado en la consola.
49 // El programa muestra el resultado en la consola.
50 // El programa muestra el resultado en la consola.
51 // El programa muestra el resultado en la consola.
52 // El programa muestra el resultado en la consola.
53 // El programa muestra el resultado en la consola.
54 // El programa muestra el resultado en la consola.
55 // El programa muestra el resultado en la consola.
56 // El programa muestra el resultado en la consola.
57 // El programa muestra el resultado en la consola.
58 // El programa muestra el resultado en la consola.
59 // El programa muestra el resultado en la consola.
60 // El programa muestra el resultado en la consola.
61 // El programa muestra el resultado en la consola.
62 // El programa muestra el resultado en la consola.
63 // El programa muestra el resultado en la consola.
64 // El programa muestra el resultado en la consola.
65 // El programa muestra el resultado en la consola.
66 // El programa muestra el resultado en la consola.
67 // El programa muestra el resultado en la consola.
68 // El programa muestra el resultado en la consola.
69 // El programa muestra el resultado en la consola.
70 // El programa muestra el resultado en la consola.
71 // El programa muestra el resultado en la consola.
72 // El programa muestra el resultado en la consola.
73 // El programa muestra el resultado en la consola.
74 // El programa muestra el resultado en la consola.
75 // El programa muestra el resultado en la consola.
76 // El programa muestra el resultado en la consola.
77 // El programa muestra el resultado en la consola.
78 // El programa muestra el resultado en la consola.
79 // El programa muestra el resultado en la consola.
80 // El programa muestra el resultado en la consola.
81 // El programa muestra el resultado en la consola.
82 // El programa muestra el resultado en la consola.
83 // El programa muestra el resultado en la consola.
84 // El programa muestra el resultado en la consola.
85 // El programa muestra el resultado en la consola.
86 // El programa muestra el resultado en la consola.
87 // El programa muestra el resultado en la consola.
88 // El programa muestra el resultado en la consola.
89 // El programa muestra el resultado en la consola.
90 // El programa muestra el resultado en la consola.
91 // El programa muestra el resultado en la consola.
92 // El programa muestra el resultado en la consola.
93 // El programa muestra el resultado en la consola.
94 // El programa muestra el resultado en la consola.
95 // El programa muestra el resultado en la consola.
96 // El programa muestra el resultado en la consola.
97 // El programa muestra el resultado en la consola.
98 // El programa muestra el resultado en la consola.
99 // El programa muestra el resultado en la consola.
100 // El programa muestra el resultado en la consola.
```

- Ingenio $\xrightarrow{\text{profesor}}$ código en repositorio
- Repositorio del curso $\xrightarrow{\text{alumno}}$ propio
- Poner en práctica: **re-utilizar** código



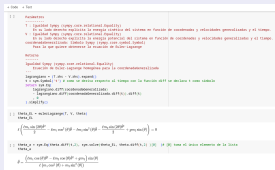
Docentes y alumnos pueden utilizar mejor su tiempo

Licklider (1957): 85 % de “pensar” es lo mundano (calcular, dibujar, etc.)



Aula y práctica: transcripción y reiteración

- Memoria $\xrightarrow{\text{profesor}}$ pizarrón/presentación
- Pizarrón/presentación $\xrightarrow{\text{alumno}}$ cuaderno
- Práctica: **reiterar** diagramas, cálculos, etc.
- Aburrimiento $\implies \downarrow$ concentración



- Ingenio $\xrightarrow{\text{profesor}}$ código en repositorio
- Repositorio del curso $\xrightarrow{\text{alumno}}$ propio
- Poner en práctica: **re-utilizar** código
- Modificarle resuelve diversas problemáticas



Los futuros ingenieros deben poder escribir código



Los futuros ingenieros deben poder escribir código

- Usan calculadora pues **aprendieron** aritmética en la primaria.



Los futuros ingenieros deben poder escribir código

- Usan calculadora pues **aprendieron** aritmética en la primaria.
- Usarán álgebra computacional pues **aprobaron** álgebra y análisis.

```
[14]: sistemaEcuaciones = [
      x_EL,
      phi_EL,
    ]
    variablesDespeje = [x.diff(t,2), phi.diff(t,2)] # despejar aceleraciones generalizadas
    variablesDespeje_sol= sym.nonlinsolve(sistemaEcuaciones, variablesDespeje ).args[0]

[15]: x_pp = sym.tq(variablesDespeje[0], variablesDespeje_sol.args[0]) # [m s^-2]
      phi_pp = sym.Eq(variablesDespeje[1], variablesDespeje_sol.args[1]) # [m s^-2]
      x_pp, phi_pp

[15]: 
$$\ddot{x} = \frac{-\ell' g m_2 \sin(\phi) + \frac{\ell' m_2 (\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{m_1 + m_2 \sin^2(\phi)}}{\ell' m_2 \cos(\phi)}, \quad \ddot{\phi} = -\frac{(\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{\ell' (m_1 + m_2 \sin^2(\phi))}$$

```



Los futuros ingenieros deben poder escribir código

- Usan calculadora pues **aprendieron** aritmética en la primaria.
- Usarán álgebra computacional pues **aprobaron** álgebra y análisis.
 - ▶ Enfocarse en nuevas habilidades, no en cálculos automatizables.

```
[14]: sistemaEcuaciones = [  
      x_EL,  
      phi_EL,  
  ]  
variablesDespeje = [x.diff(t,2), phi.diff(t,2)] # despejar aceleraciones generalizadas  
variablesDespeje_sol= sym.nonlinsolve(sistemaEcuaciones, variablesDespeje ).args[0]  
  
[15]: x_pp = sym.tq(variablesDespeje[0], variablesDespeje_sol.args[0] ) # [m s-2]  
      phi_pp = sym.Eq(variablesDespeje[1], variablesDespeje_sol.args[1] ) # [m s-2]  
      x_pp, phi_pp  
  
[15]: 
$$\ddot{x} = \frac{-\ell' g m_2 \sin(\phi) + \frac{\ell' m_2 (\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{m_1 + m_2 \sin^2(\phi)}}{\ell' m_2 \cos(\phi)}, \quad \ddot{\phi} = -\frac{(\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{\ell' (m_1 + m_2 \sin^2(\phi))}$$

```



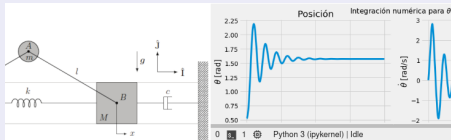
Los futuros ingenieros deben poder escribir código

- Usan calculadora pues **aprendieron** aritmética en la primaria.
- Usarán álgebra computacional pues **aprobaron** álgebra y análisis.
 - ▶ Enfocarse en nuevas habilidades, no en cálculos automatizables.
 - ▶ Con cálculo numérico resolverán lo imposible en pizarrón/papel.

```
[14]: sistemaEcuaciones = [  
    x_EL,  
    phi_EL,  
]  
variablesDespeje = [x.diff(t,2), phi.diff(t,2)] # despejar aceleraciones generalizadas  
variablesDespeje_sol= sym.nonlinsolve(sistemaEcuaciones, variablesDespeje ).args[0]
```

```
[15]: x_pp = sym.Eq(variablesDespeje[0], variablesDespeje_sol.args[0]) # [m s-2]  
phi_pp = sym.Eq(variablesDespeje[1], variablesDespeje_sol.args[1]) # [m s-2]  
x_pp, phi_pp
```

$$\ddot{x} = \frac{-\ell' g m_2 \sin(\phi) + \frac{\ell' m_2 (\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{m_1 + m_2 \sin^2(\phi)}}{\ell' m_2 \cos(\phi)}, \quad \ddot{\phi} = -\frac{(\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{\ell' (m_1 + m_2 \sin^2(\phi))}$$

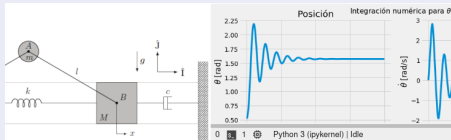


Los futuros ingenieros deben poder escribir código

- Usan calculadora pues **aprendieron** aritmética en la primaria.
- Usarán álgebra computacional pues **aprobaron** álgebra y análisis.
 - ▶ Enfocarse en nuevas habilidades, no en cálculos automatizables.
 - ▶ Con cálculo numérico resolverán lo imposible en pizarrón/papel.

```
[14]: sistemaEcuaciones = [  
    x_EL,  
    phi_EL,  
]  
variablesDespeje = [x.diff(t,2), phi.diff(t,2)] # despejar aceleraciones generalizadas  
variablesDespeje_sol= sym.nonlinsolve(sistemaEcuaciones, variablesDespeje ).args[0]  
  
[15]: x_pp = sym.Eq(variablesDespeje[0], variablesDespeje_sol.args[0]) # [m s-2]  
phi_pp = sym.Eq(variablesDespeje[1], variablesDespeje_sol.args[1]) # [m s-2]  
x_pp, phi_pp  
  
[15]: 
$$\ddot{x} = \frac{-\ell' g m_2 \sin(\phi) + \frac{\ell' m_2 (\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{m_1 + m_2 \sin^2(\phi)}}{\ell' m_2 \cos(\phi)}, \quad \ddot{\phi} = -\frac{(\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{\ell' (m_1 + m_2 \sin^2(\phi))}$$

```

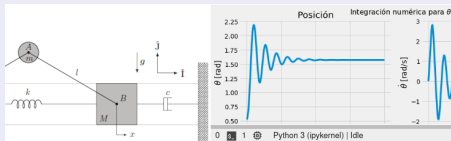


Los futuros ingenieros deben poder escribir código

- Usan calculadora pues **aprendieron** aritmética en la primaria.
- Usarán álgebra computacional pues **aprobaron** álgebra y análisis.
 - ▶ Enfocarse en nuevas habilidades, no en cálculos automatizables.
 - ▶ Con cálculo numérico resolverán lo imposible en pizarrón/papel.

```
[14]: sistemaEcuaciones = [  
    x_EL,  
    phi_EL,  
]  
variablesDespeje = [x.diff(t,2), phi.diff(t,2)] # despejar aceleraciones generalizadas  
variablesDespeje_sol= sys.nonlinearize(sistemaEcuaciones, variablesDespeje ).args[0]  
  
[15]: x_pp = sym.Eq(variablesDespeje[0], variablesDespeje_sol.args[0]) # [m s-2]  
phi_pp = sym.Eq(variablesDespeje[1], variablesDespeje_sol.args[1]) # [m s-2]  
x_pp, phi_pp  
  
[15]: 
$$\ddot{x} = \frac{-\ell' g m_2 \sin(\phi) + \frac{\ell' m_2 (\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{m_1 + m_2 \sin^2(\phi)}}{\ell' m_2 \cos(\phi)}, \quad \ddot{\phi} = -\frac{(\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{\ell' (m_1 + m_2 \sin^2(\phi))}$$

```



Papert (1980) “El aprendizaje sucede cuando el alumno toma las riendas”

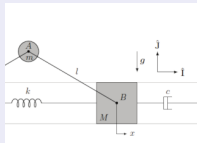
- El código inicial es provisto por el docente.

Los futuros ingenieros deben poder escribir código

- Usan calculadora pues **aprendieron** aritmética en la primaria.
- Usarán álgebra computacional pues **aprobaron** álgebra y análisis.
 - ▶ Enfocarse en nuevas habilidades, no en cálculos automatizables.
 - ▶ Con cálculo numérico resolverán lo imposible en pizarrón/papel.

```
[14]: sistemaEcuaciones = [  
    x_EL,  
    phi_EL,  
]  
variablesDespeje = [x.diff(t,2), phi.diff(t,2)] # despejar aceleraciones generalizadas  
variablesDespeje_sol= sys.nonlinearize(sistemaEcuaciones, variablesDespeje ).args[0]  
  
[15]: x_pp = sym.Eq(variablesDespeje[0], variablesDespeje_sol.args[0]) # [m s-2]  
phi_pp = sym.Eq(variablesDespeje[1], variablesDespeje_sol.args[1]) # [m s-2]  
x_pp, phi_pp  
  
[15]: 
$$\ddot{x} = \frac{-\ell' g m_2 \sin(\phi) + \frac{\ell' m_2 (\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{m_1 + m_2 \sin^2(\phi)}}{\ell' m_2 \cos(\phi)}, \quad \ddot{\phi} = -\frac{(\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{\ell' (m_1 + m_2 \sin^2(\phi))}$$

```



Papert (1980) “El aprendizaje sucede cuando el alumno toma las riendas”

- El código inicial es provisto por el docente.
- Modificaciones aditivas resuelven nuevas problemáticas.

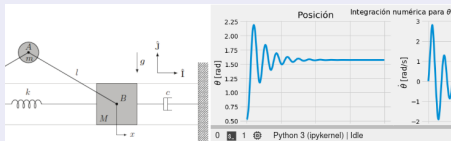


Los futuros ingenieros deben poder escribir código

- Usan calculadora pues **aprendieron** aritmética en la primaria.
- Usarán álgebra computacional pues **aprobaron** álgebra y análisis.
 - ▶ Enfocarse en nuevas habilidades, no en cálculos automatizables.
 - ▶ Con cálculo numérico resolverán lo imposible en pizarrón/papel.

```
[14]: sistemaEcuaciones = [  
    x_EL,  
    phi_EL,  
]  
variablesDespeje = [x.diff(t,2), phi.diff(t,2)] # despejar aceleraciones generalizadas  
variablesDespeje_sol= sys.nonlinearize(sistemaEcuaciones, variablesDespeje ).args[0]  
  
[15]: x_pp = sym.Eq(variablesDespeje[0], variablesDespeje_sol.args[0]) # [m s-2]  
phi_pp = sym.Eq(variablesDespeje[1], variablesDespeje_sol.args[1]) # [m s-2]  
x_pp, phi_pp  
  
[15]: 
$$\ddot{x} = \frac{-\ell' g m_2 \sin(\phi) + \frac{\ell' m_2 (\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{m_1 + m_2 \sin^2(\phi)}}{\ell' m_2 \cos(\phi)}, \quad \ddot{\phi} = -\frac{(\ell' m_2 \cos(\phi) \dot{\phi}^2 + g m_1 + g m_2) \sin(\phi)}{\ell' (m_1 + m_2 \sin^2(\phi))}$$

```



Papert (1980) “El aprendizaje sucede cuando el alumno toma las riendas”

- El código inicial es provisto por el docente.
- Modificaciones aditivas resuelven nuevas problemáticas.
- Alumno se torna autónomo reutilizando el propio.

Todo el material es editable en línea



Todo el material es editable en línea

Cuaderno programable en línea: texto + ecuaciones + código

File Edit View Run Kernel Tabs Settings Help

PÉNDULOENHEBRADOSOLVED.IPYNB

Launcher cursoJupyter.Ipynb pénduloEnhebradoSolved.Ipy × Python 3

3. Obtenga una expresión para la tensión que ejerce la barra

$$Q_d = \lambda_1 \frac{\partial f_1}{\partial d} = \lambda_1$$

Por tanto hay que resolver el sistema con las 3 ecuaciones de Euler-Lagrange y la única de ligadura para determinar λ_1 . Esta última hay que resolverla para su caso homogéneo y expresar su derivada segunda para que esté en el mismo orden que las de Euler-Lagrange, a fin de cuentas estamos resolviendo sistemas diferenciales de 2.º orden.

```
[14]: f_1
```

[14]: $f_1 = -l + d$

Determinamos también $\ddot{\theta}_1$ y $\ddot{\theta}_2$ pues serán necesarias para los cálculos numéricos posteriores.

```
[15]: sistema = [theta1_EL.expand(),
               theta2_EL.expand(),
               d_EL.expand(),
               sym.Eq(f_1.rhs.diff(t,2), 0)], # esto es igual a d punto punto = 0
variables = [theta1.diff(t,2), theta2.diff(t,2), lambda_1]
variables_sol = sym.nonlinsolve(sistema, variables).args[0]
```

```
[16]: lambda_1_sol = sym.Eq(lambda_1, variables_sol.args[2])
      lambda_1_sol.simplify()
```

```
[16]:
```

$$\lambda_1 = \frac{m \left(2a \cos(\theta_1 - \theta_2) \dot{\theta}_1^2 + g \cos(2\theta_1 - \theta_2) + g \cos(\theta_2) + 2d\ddot{\theta}_2 - 2\ddot{d} \right)}{\cos(2\theta_1 - 2\theta_2) - 3}$$

Simple 0 2 Python 3 | Idle Saving completed Mode: Command Ln 1, Col 1 pénduloEnhebradoSolved.Ipynb

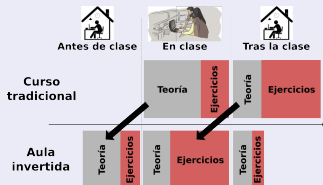


Trabajo sincrónico y asincrónico sobre el código

Teoría y ejercicios resueltos en línea en cuadernos programables

- Modificando su código se resuelven guías semanales de ejercicios.

Aula invertida



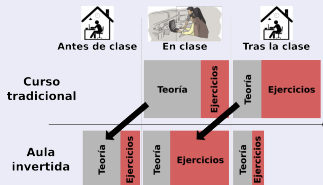
Sincrónico	Teoría	Ejercicios
Antes	Leer y aplicar	Iniciarlos
Durante	Aclarar dudas	Terminarles
Luego	Consultas adicionales	Correcciones del docente

Trabajo sincrónico y asincrónico sobre el código

Teoría y ejercicios resueltos en línea en cuadernos programables

- Modificando su código se resuelven guías semanales de ejercicios.
- Asincrónico: consultas en línea 24/7 **públicas** hacia otros alumnos.

Aula invertida



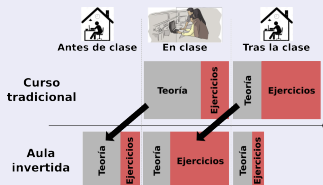
Sincrónico	Teoría	Ejercicios
Antes	Leer y aplicar	Iniciarlos
Durante	Aclarar dudas	Terminarles
Luego	Consultas adicionales	Correcciones del docente

Trabajo sincrónico y asincrónico sobre el código

Teoría y ejercicios resueltos en línea en cuadernos programables

- Modificando su código se resuelven guías semanales de ejercicios.
- Asincrónico: consultas en línea 24/7 **públicas** hacia otros alumnos.
- Cuadernos multi-usuario: resolución en colaboración remota.

Aula invertida



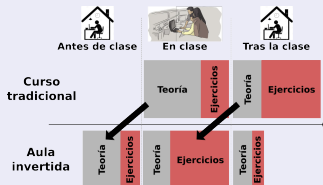
Sincrónico	Teoría	Ejercicios
Antes	Leer y aplicar	Iniciarlos
Durante	Aclarar dudas	Terminarles
Luego	Consultas adicionales	Correcciones del docente

Trabajo sincrónico y asincrónico sobre el código

Teoría y ejercicios resueltos en línea en cuadernos programables

- Modificando su código se resuelven guías semanales de ejercicios.
- Asincrónico: consultas en línea 24/7 **públicas** hacia otros alumnos.
- Cuadernos multi-usuario: resolución en colaboración remota.
- Sincrónico: **docente disponible** al finalizar ejercicios

Aula invertida



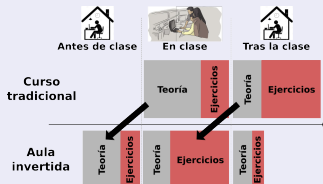
Sincrónico	Teoría	Ejercicios
Antes	Leer y aplicar	Iniciarles
Durante	Aclarar dudas	Terminarles
Luego	Consultas adicionales	Correcciones del docente

Trabajo sincrónico y asincrónico sobre el código

Teoría y ejercicios resueltos en línea en cuadernos programables

- Modificando su código se resuelven guías semanales de ejercicios.
- Asincrónico: consultas en línea 24/7 **públicas** hacia otros alumnos.
- Cuadernos multi-usuario: resolución en colaboración remota.
- Sincrónico: **docente disponible** al finalizar ejercicios
- Entrega semanal para corrección **semanal** y **obligatoria**.


Aula invertida



Sincrónico	Teoría	Ejercicios
Antes	Leer y aplicar	Iniciarlos
Durante	Aclarar dudas	Terminarles
Luego	Consultas adicionales	Correcciones del docente

Asistencia docente y corrección asincrónica

Google Colaboratory: comentando y editando el ejercicio del alumno

 07 No conservativas | ej4 ☆

Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda Se editó por última vez: 3 de junio

+ Código + Texto

```
[ ] # Energía potencial
m1_V = - (m1* g* (- N.y)).dot(m1_r)
# pot_k1 = unMedio* (-k1* ((l10 + x1)* (sym.cos(theta) - sym.sin(theta)) )**2 ) # mal
pot_k1 = unMedio* k1* (l10 + x1)**2 # Lo escribí yo
# pot_k2 = unMedio* -k2* (l20 + x)**2
pot_k2 = unMedio* k2* (l20 + x)**2
V = sym.Eq(sym.Symbol('V'), m1_V + pot_k1 + pot_k2 ) #agrega el potencial elastico k en la ecuacion
V
```

$$V = gm_1(-l_{10} - x_1)\sin(\theta) + \frac{k_1(l_{10} + x_1)^2}{2} + \frac{k_2(l_{20} + x)^2}{2}$$


▼ Lagrangiano


```
[ ] L = sym.Eq(sym.Symbol('\mathcal{L}'), (T.rhs - V.rhs))
L
```

$$\mathcal{L} = -gm_1(-l_{10} - x_1)\sin(\theta) - \frac{k_1(l_{10} + x_1)^2}{2} - \frac{k_2(l_{20} + x)^2}{2} + \frac{(m_0 + m_1)(2\cos(\theta)\dot{x}_1 + \dot{x}^2 + \dot{x}_1^2)}{2}$$

ECUACIONES DE EULER

Para x

 Victor Alexis Bettachini Resolver 31 de may. de 2021 (editado el 31 de may. de 2021)
- El estiramiento del resorte de k_1 es colineal con x_1 . No tienen sentido pensar en proyecciones (si es lo que hiciste, que realmente no entiendo).
- ¿Porque negativos los k?

 Victor Alexis Bettachini Resolver 31 de may. de 2021



Seguimiento individualizado

Registro del cumplimiento con entregas semanales

Calificaciones

Vencimiento el 28 sept

Buscar alumnos

	g06e03 28 sept	g06e04 28 sept	g06e05 28 sept	g05e01a 14 sept	g05e01c 14 sept	g05e02 14 sept	g05e03 14 sept
Promedio de clase							
[Avatar] GARCIA, A.	Visto		Visto	Devuelto	Entregado	Entregado	Entregado
[Avatar] CHACABARTO, L.				Devuelto	Entregado	Entregado	Entregado
[Avatar] CORTES, M.				Entregado	Entregado	Entregado	Entregado
[Avatar] CONTRERAS, J.	Visto	Visto	Visto	Entregado	Entregado	Entregado	Entregado
[Avatar] DIAZ, A.				Entregado	Entregado		Entregado
[Avatar] FERRAZ, L.	Visto	Visto	Visto	Entregado	Entregado	Visto	Entregado
[Avatar] RODRIGUEZ, J.	Visto			Entregado	Entregado	Entregado	Entregado



Resumiendo

Curso centrado en código

- Teoría: texto + ecuaciones + código ejecutable en cuadernos digitales.

Modalidad de aula invertida



Resumiendo

Curso centrado en código

- Teoría: texto + ecuaciones + código ejecutable en cuadernos digitales.
- Práctica: reutilización del código del docente.

Modalidad de aula invertida



Resumiendo

Curso centrado en código

- Teoría: texto + ecuaciones + código ejecutable en cuadernos digitales.
- Práctica: reutilización del código del docente.
- Ejecución en línea:

Modalidad de aula invertida



Curso centrado en código

- Teoría: texto + ecuaciones + código ejecutable en cuadernos digitales.
- Práctica: reutilización del código del docente.
- Ejecución en línea:
 - ▶ Colaboración y corrección remota.

Modalidad de aula invertida



Curso centrado en código

- Teoría: texto + ecuaciones + código ejecutable en cuadernos digitales.
- Práctica: reutilización del código del docente.
- Ejecución en línea:
 - ▶ Colaboración y corrección remota.
 - ▶ No requiere computadoras en el campus.

Modalidad de aula invertida



Resumiendo

Curso centrado en código

- Teoría: texto + ecuaciones + código ejecutable en cuadernos digitales.
- Práctica: reutilización del código del docente.
- Ejecución en línea:
 - ▶ Colaboración y corrección remota.
 - ▶ No requiere computadoras en el campus.
 - ▶ Registro fechado del trabajo del alumno.

Modalidad de aula invertida



Resumiendo

Curso centrado en código

- Teoría: texto + ecuaciones + código ejecutable en cuadernos digitales.
- Práctica: reutilización del código del docente.
- Ejecución en línea:
 - ▶ Colaboración y corrección remota.
 - ▶ No requiere computadoras en el campus.
 - ▶ Registro fechado del trabajo del alumno.

Modalidad de aula invertida

- Teoría: énfasis en la lectura autónoma por parte del alumno.



Curso centrado en código

- Teoría: texto + ecuaciones + código ejecutable en cuadernos digitales.
- Práctica: reutilización del código del docente.
- Ejecución en línea:
 - ▶ Colaboración y corrección remota.
 - ▶ No requiere computadoras en el campus.
 - ▶ Registro fechado del trabajo del alumno.

Modalidad de aula invertida

- Teoría: énfasis en la lectura autónoma por parte del alumno.
- Consultas asincrónicas públicas.



Curso centrado en código

- Teoría: texto + ecuaciones + código ejecutable en cuadernos digitales.
- Práctica: reutilización del código del docente.
- Ejecución en línea:
 - ▶ Colaboración y corrección remota.
 - ▶ No requiere computadoras en el campus.
 - ▶ Registro fechado del trabajo del alumno.

Modalidad de aula invertida

- Teoría: énfasis en la lectura autónoma por parte del alumno.
- Consultas asincrónicas públicas.
- Práctica: Docente asiste personalmente cuando más se lo requiere, para finalizar ejercicios.



Actualidad del proyecto



2023 Retro-alimentación de los alumnos mejoró:



2023 Retro-alimentación de los alumnos mejoró:

- Apuntes y código en el repositorio.



2023 Retro-alimentación de los alumnos mejoró:

- Apuntes y código en el repositorio.
 - Metodología ejercitación y evaluación.
- Mayor exigencia de ejercicios → mejor respuesta.



2023 Retro-alimentación de los alumnos mejoró:

- Apuntes y código en el repositorio.
 - Metodología ejercitación y evaluación.
- Mayor exigencia de ejercicios → mejor respuesta.

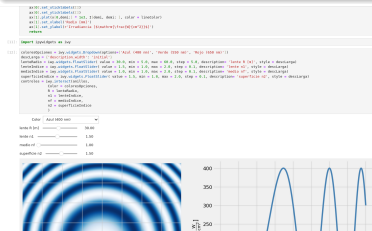


Actualidad del proyecto

2023 Retro-alimentación de los alumnos mejoró:

- Apuntes y código en el repositorio.
 - Metodología ejercitación y evaluación.
- Mayor exigencia de ejercicios → mejor respuesta.

2024 • Física II empleará simulaciones provistas por nosotros.



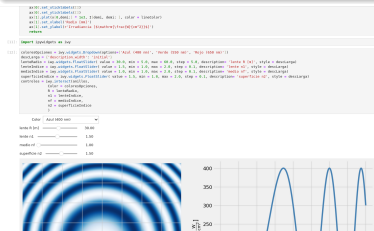
Actualidad del proyecto

2023 Retro-alimentación de los alumnos mejoró:

- Apuntes y código en el repositorio.
 - Metodología ejercitación y evaluación.
- Mayor exigencia de ejercicios → mejor respuesta.

2024

- Física II empleará simulaciones provistas por nosotros.
- *Prompt engineering*: alumnos generarán código con IA.



```
lagrangiano = (T.rhs - V.rhs).expand()  
t = sym.Symbol('t') # como se deriva respecto al tiempo con la función diff se declara t como simbolo  
return sym.Eq(  
    lagrangiano.diff(coordenadaGeneralizada).diff(t).diff(t)  
    - lagrangiano.diff(coordenadaGeneralizada).diff(t)  
    , 0  
)  
.simplify()
```

```
x1_EL = eulerLagrange(T, V, x1)  
x1_EL
```

```
...  
m1*M2*x1**2/2 - g*m1 + g*m2 + m1*x1 + m2*x1 = 0
```

Esta es una ecuación diferencial lineal de segundo orden homogenea. De aquí se puede despejar \ddot{x}

```
#Despejar x1Punto  
x1PuntoPunto = sym.solve(t, x1.diff(t, t)).args[0]
```

