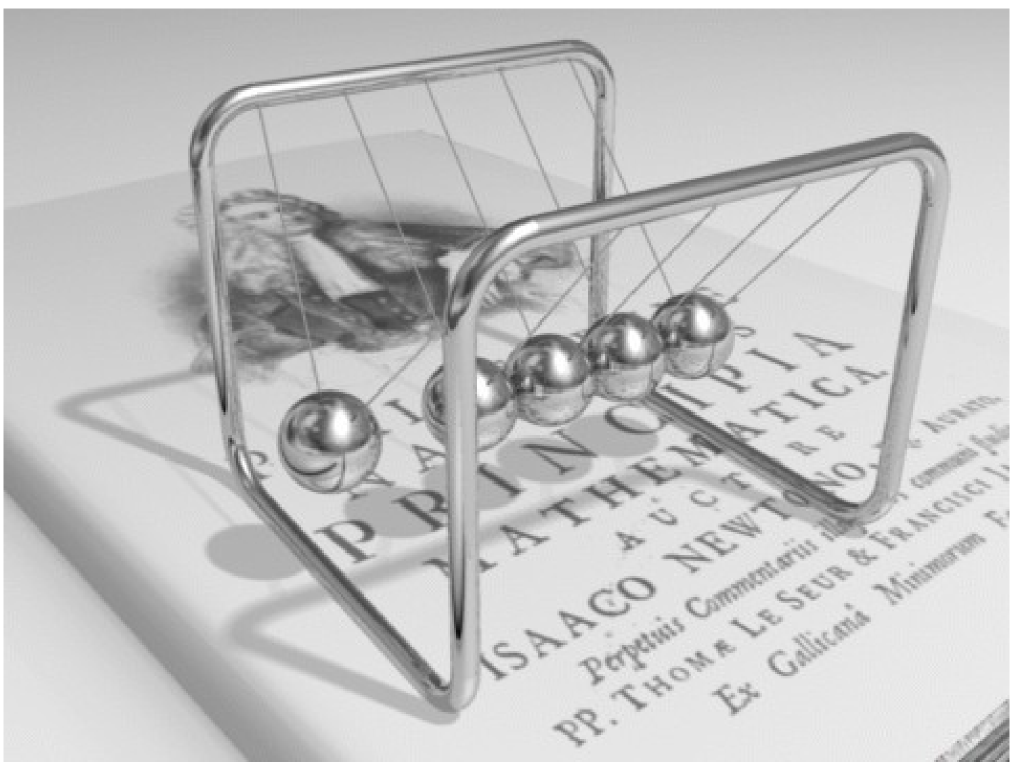
Proyecto final de Informática

Tema:
Conservación de la energía mecánica



Estudiante: Stella Christian
Año: 2023

La física computacional y la conservación de la energía mecánica

Resumen:

Este trabajo corresponde al Proyecto Final de la materia Informática (Física computacional) de la Licenciatura en física de la UNLPAM, del año 2023.

El objetivo es aplicar un programa que permita la resolución de problemas de una forma ordenada, rápida, fácil y efectiva en algún área específica de las ciencias físicas. En este caso usamos como punto de partida un principio fundamental: ***La conservación de la energía mecánica.***

El trabajo consta de la siguiente estructura:

(1) Una introducción donde encontraremos una breve explicación de este principio, su importancia en la física y los conceptos principales que lo conforman.

(2) El desarrollo, que estará dividido en 3 secciones:
- **a)** Una breve nota histórica sobre la formulación de este principio.
- **b)** Definiremos distintos tipos de energía, el trabajo y la forma de calcularlos matemáticamente.
- **c)** Presentaremos la herramienta computacional, su contenido, funcionamiento y cómo podría ser útil en algún área determinada de la vida cotidiana o de algún proyecto en particular.

(3) Luego obtendremos alguna conclusión que confirme o no, nuestras expectativas y motivaciones iniciales

(4) Finalmente incluiremos las palabras clave, las referencias bibliográficas y un anexo donde estará presente el código del programa realizado en lenguaje Python.

1) Introducción:

Importancia de la Conservación de la Energía Mecánica

Hasta donde sabemos existen tres cantidades que siempre se conservan: el momento lineal, el momento angular y la energía. Estas tres leyes de la conservación son fundamentales para cualquier estudiante de física y encierran propiedades de la naturaleza que sostienen nuestro paradigma actual del universo y que data de hace más de 200 años. Como dice Paul Hewitt: "(...)El estudio de las diversas formas de energía y sus transformaciones entre sí ha conducido a una de las grandes generalizaciones de la física: la ley de la conservación de la energía:

La energía no se puede crear ni destruir; se puede transformar de una forma a otra, pero la cantidad total de energía nunca cambia.

Cuando examinamos cualquier sistema en su totalidad, sea tan sencillo como un péndulo que oscila o tan complejo como una supernova que explota, hay una cantidad que no se crea ni se destruye: la energía. Puede cambiar de forma, o tan sólo se puede transferir de un lugar a otro; pero hasta donde sabemos, la cuenta total de la energía permanece igual(...)"[1]

La conservación de la energía mecánica es uno de los principios más importantes de la Física. Una de las reliquias de nuestro conocimiento colectivo. Esta ley establece que la energía mecánica de un sistema cerrado siempre permanece constante. Dentro de este sistema puede haber distintas formas de energía que se van transformando unas en otras por la interacción de las fuerzas internas que actúan en él. Mas la única manera de que se modifique la cantidad de energía mecánica de ese sistema es a partir de fuerzas externas que realizan un trabajo sobre dicho sistema. Es decir que hay otro concepto fundamental que debemos abordar: El teorema del trabajo y la energía, que establece que el trabajo neto de todas las fuerzas sobre un objeto es igual a la variación de su energía cinética. Y la velocidad con la que se realiza un trabajo, o que se transfiere energía, se denomina potencia, que la vemos reflejada cotidianamente en los KiloWatts de la empresa de electricidad o los caballos de fuerza de una moto. Dentro de la energía mecánica hay dos tipos de energías, la cinética y la potencial, que se vinculan y equilibran constantemente, donde la energía cinética se transforma en potencial y viceversa.

Ahora bien, de este principio se deducen fórmulas para distintas áreas que resultan ser herramientas imprescindibles para cualquier ciencia aplicada o ingeniería. Una de las grandes ventajas de este modelo es que es una manera más sencilla de resolver problemas físicos y obtener los mismos resultados que con otros modelos un poco más complejos. Incluso podemos analizar deportes, artes, oficios y muchas otras disciplinas humanas; podemos obtener información valiosa para entender mejor la naturaleza, nuestro organismo, nuestras actividades y nuestra sociedad en general e infinidad de fenómenos que, siendo coherente con este principio, son el punto de partida de nuevas teorías o tecnologías.

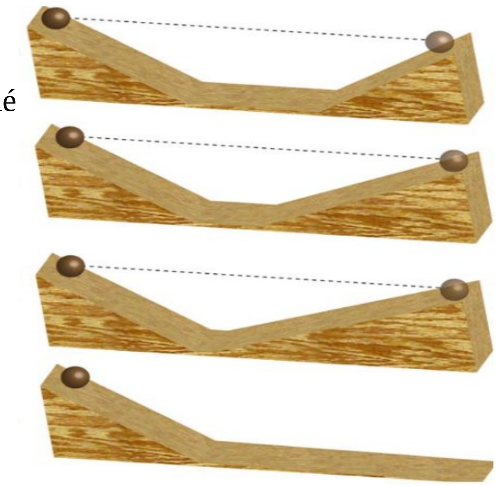
Por mencionar algunos ejemplos clásicos tenemos el principio de Bernoulli y la fórmula de Torricelli en el área de dinámica de fluidos, que es fundamental para el desarrollo de las redes pluviales. Lo mismo sucede con la ley de Ohm y las leyes de Kirchhoff en circuitos eléctricos, y la ley de Maxwell-Faraday, que junto a la dinámica de fluidos nos abren la puerta para que podamos transformar la energía eólica e hídrica en eléctrica. El uso de estos principios para estudiar choques y colisiones, fuerzas de impacto o distancias de frenado son fundamentales para la industria automotriz y las políticas viales. Con estos cálculos se desarrollan los vehículos (ruedas, frenos, alternadores, cinturones de seguridad) y los caminos que transitan éstos (diseño de curvas, pendientes, cartelería, etc etc y etc.).

1 Paul Hewitt, Física conceptual. Página 117, Capítulo 7: Energía. 10ma edición, 2007

No es es exagerado afirmar que el principio de la conservación de la energía mecánica y el teorema del trabajo y la energía cinética conforman uno de los pilares de nuestra civilización moderna y que toda nuestra sociedad los atraviesa sin esquivarlos.

2) Desarrollo:

<u>a) Breve historia del modelo</u>



La pregunta esencial es: si la energía siempre se conserva, ¿por qué los músculos se cansan y por qué los pesos se caen? ¿por qué la altura obtenida en un salto es proporcional a la velocidad alcanzada, si tenemos la misma masa?

Hacia finales del siglo XVI Galileo Galilei se hizo una pregunta semejante. Fue una pregunta a la cual ni el propio Galileo pudo encontrar respuesta. Sin embargo, mientras utilizaba planos inclinados para disimular la aceleración de los cuerpos al caer, descubrió otra cosa: cualquiera que fuera el camino seguido, la pelota recuperaba su altura inicial, como si recordara su posición inicial.

Por supuesto Galileo sabía que un objeto no podía recordar dónde había estado pero comprendió que la pelota conservaba algo muy poderoso. Si no era memoria ¿qué conservaba la pelota? La respuesta no vino ráppido, pero la velocidad es la clave de la respuesta. Comenzando desde la misma altura, con independencia de la inclinación del plano, cuando la pelota llega al punto más bajo, lo hace siempre con la misma velocidad. La energía que tenía la pelota a su altura original se conserva mientras rueda transformada en velocidad.

Los 250 años siguientes fueron testigos de importantes y fundamentales aportes, de Newton y Leibniz por supuesto, pero también de Torricelli, Descartes, Halley, Kepler y otros que no vamos a detallar acá, pero que le dieron forma a todo lo que hoy conocemos como “Física clásica”, incluyendo un gran acercamiento al principio de la conservación de la energía. **Es difícil establecer una fecha concreta para la inclusión de este principio al modelo físico de la época. Durante esos 200 años se utilizaban expresiones similares, se respiraba en el aire que la energía se conservaba, pero sin que esté cristalizada en una teoría acabada y propia. Podemos adjudicarle la formalización de estos conceptos a** *Von Mayer en 1842 con su trabajo “Observaciones sobre las fuerzas inorgánicas de la naturaleza”, a James Joule en 1843 o a partir de Julio de* 1847 con *Hermann von Helmholtz.* Este médico alemán que estaba estudiando el metabolismo del músculo, tituló su trabajo ”Über die Erhaltung der Kraft” (Sobre la conservación de la fuerza usando “fuerza” en el sentido moderno de “energía”). En él plantea “(...)Llegamos a la conclusión de que la Naturaleza en su conjunto posee una reserva de fuerza [energía] que no puede de ninguna manera ser a*umentada ni disminuida y que, por lo tanto, la cantidad de fuerza en la Naturaleza es igual de eterna e inalterable que la cantidad de materia. Expresado en esta forma he llamado a la ley general «El Principio de la Conservación de la Fuerza». (...)”* [2]

Über

die Erhaltung der Kraft,

eine physikalische Abhandlung,

vorgetragen in der Sitzung der physikalischen Gesellschaft zu Berlin am 23sten Juli 1847

von

Dr. H. Helmholtz.

Berlin,

Druck und Verlag von G. Reimer.

1847.

---

2 Jhon Bolaños, “Análisis histórico del principio de conservación de la energía, algunas pautas en la enseñanza de una ley”, páginas 25-53.

Lo cierto es que desde 1850 la conservación de la energía mecánica es formalmente parte de nuestro conocimiento científico producido por el esfuerzo colectivo de una gran cantidad de científicos y científicas que cambiaron el paradigma de su época, heredado de la antigua filosofía griega.

b) *Energía y trabajo*

Trabajo:

Una definición formal del trabajo es que es un *mecanismo de transferencia de energía por medios mecánicos* pero se define matemáticamente como el producto escalar entre los vectores fuerza y desplazamiento.

Es decir que para una fuerza constante y un desplazamiento rectilíneo, el trabajo que realiza la fuerza es:

$$W = F \cdot \Delta x$$
$$W = F\, \Delta x \cos\theta$$

Asimismo podemos definir a la energía como l*a capacidad de realizar un trabajo,* y que se manifiesta de muchas maneras. Las que vamos a considerar en este caso son las siguientes:

Energía potencial:

Esta energía está asociada a la posición de los cuerpos en un sistema, que interactúan entre sí mediante fuerzas conservativas (el trabajo que realiza una fuerza conservativa es independiente de la trayectoria).

Este tipo de energía es una medida del potencial o de la posibilidad de efectuar un trabajo. Dentro de esta categoría utilizaremos dos tipos: *la potencial gravitatoria y la potencial elástica*.

Al levantar una roca existe la posibilidad de que la fuerza de gravedad realice trabajo sobre ella, pero sólo si la roca se deja caer al suelo. O si comprimimos un resorte éste va a realizar un trabajo si se suelta. Por ello la energía asociada con la posición se llama *energía potencial.*

Energía potencial gravitatoria:

Cuanto mayor es el peso, mayor fuerza se necesita para levantarlo. Y por supuesto, cuanto más altura, más trabajo. Dijimos que el trabajo es igual a fuerza por desplazamiento, que en este caso es una altura. Cuando el trabajo se realiza cerca de la superficie de la tierra, la fuerza en esta ecuación es la fuerza constante de la gravedad, que es igual a masa por la aceleración de la gravedad. Entonces, el trabajo que se necesita para aumentar la energía potencial gravitatoria es igual a la masa por gravedad y por altura:

$$\mathbf{W} = mgy.$$

La fuerza g acelera la masa hacia la tierra, por lo tanto, la fuerza opuesta se utiliza para vencer la gravedad al levantar el peso a una cierta altura. *En la conservación de la energía, el papel que juega el trabajo es transferir energía de un lugar a otro.* Un trabajo contra una fuerza constante y opuesta.

Al levantar un bloque desde una altura a otra, tenemos que el trabajo es la diferencia de energía potencial entre las dos alturas. Es decir, el cambio en la energía potencial.

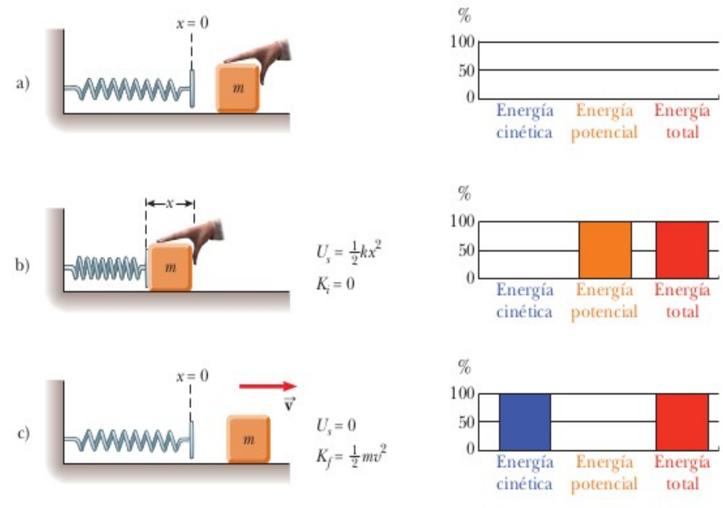
$$Ug = mgy$$
$$\mathbf{W} = \Delta Ug = mg\,(y\,f - y\,i)$$

Energía potencial elástica:

La Ley de Hooke describe la conducta elástica de los sólidos. Esta ley postula que el desplazamiento o la deformación sufrida por un elástico será directamente proporcional a la fuerza o carga que se le aplique. Donde **k** es la constante elástica y **-x** es el desplazamiento respecto del equilibrio.

$$Fs = -kx$$

Si integramos esta expresión que representa la fuerza elástica a lo largo de una distancia, obtenemos el trabajo producido por esta fuerza, la cual es:



$$Ue = \tfrac{1}{2}\, kx^2$$
$$\mathbf{W} = \Delta Ue = (\tfrac{1}{2}\, kx_f^2) - (\tfrac{1}{2}\, kx_i^2)$$

De esto podemos deducir algo importante: ***La variación de energía potencial se calcula como el negativo del trabajo que realiza la fuerza conservativa:***

$$\Delta U\, Fg = -W\, g$$
$$\Delta U\, Fe = -W\, e$$

Energía cinética:

Representa la energía *asociada al movimiento*. En una trayectoria rectilínea, donde el movimiento pertenece al centro de masas, la energía cinética es traslacional, y depende de su masa y la velocidad del centro de masas. En una trayectoria circular, donde el movimiento se produce alrededor del centro de masas mientras que éste está en reposo, la energía cinética es rotacional, y depende de su momento de inercia y de su velocidad angular.

Energía cinética traslacional:

$$K\mathbf{tras} = \tfrac{1}{2}\, mv^2$$
$$W = \Delta K\mathbf{tras} = (\tfrac{1}{2}\, mv^2\mathbf{f}) - (\tfrac{1}{2}\, mv^2 i)$$

Energía cinética rotacional:

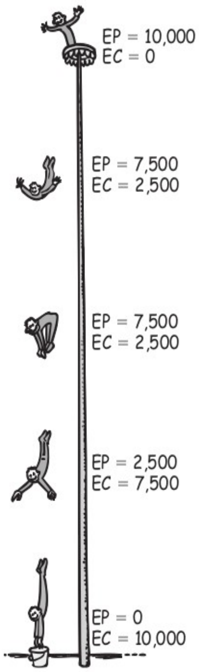
$$Krot = \tfrac{1}{2}\, Iw^2$$
$$W = \Delta K\mathbf{rot} = (\tfrac{1}{2}\, \mathbf{I}\underline{w}^2\mathbf{f}) - (\tfrac{1}{2}\, \mathbf{I}\underline{w}^2 i)$$

La ***energía cinética total*** *de un sistema de partículas se puede expresar como la suma de la energía cinética del centro de masas más la energía cinética asociada al movimiento alrededor del centro de masas*. Es decir que juntas forman la energía cinética total, aunque puede estar presente una sin la otra, dependiendo del caso.

$$Kt = (\tfrac{1}{2}\; mv_{cm}^2) + (\tfrac{1}{2}\; Iw^2)$$

De esto deducimos “El teorema del trabajo y la energía cinética” que establece lo siguiente:

El trabajo realizado por la fuerza neta que actúa sobre una partícula es igual al cambio en su energía cinética:



$$W = \Delta K = K f - K i$$

Principio de conservación de la energía mecánica:

En un sistema aislado (sin interacción con el entorno) cuyas partes interaccionan sólo mediante fuerzas conservativas, la energía mecánica permanece constante y se conserva la energía total:

$$\Delta K + \Delta U = 0$$
$$\Delta Em = 0$$
$$E\,T = K + U + E\,i = cte$$

Dónde "K" y "U" son la energía mecánica (cinética y potencial macroscópicas) y "Ei" es la energía cinética y potencial microscópicas (átomos y moléculas).

En un **sistema no aislado** (en interacción con el entorno) cuyas partes interaccionan también con fuerzas disipativas, el sistema gana o pierde energía mecánica:

Disipación de energía mecánica:

El coeficiente de fricción de una superficie y la Fuerza normal, que siempre es perpendicular a la superficie, producen una fuerza de rozamiento que se opone al deslizamiento de una partícula.

Esta es una fuerza no-conservativa que disipa la energía mecánica (generalmente en calor) y se convierte en la energía interna que hace perder energía al sistema. Es decir que si planteamos la conservación de la energía, vamos a encontrar que no es cero, sino que la energía mecánica final va a ser menor que la inicial:

$$Emf < Emi$$

Lo que sucedió es una transferencia de energía en un sistema no aislado, es decir, hubo un ***trabajo externo de la fuerza de rozamiento:***

- Se "pierde" energía mecánica → $\Delta Em < 0$
- Aumenta la energía interna del objeto en la superficie → $\Delta Eint$

Finalmente podemos decir que en un sistema no aislado el intercambio de energía por trabajo con el medio exterior es:

$$Wext = \Delta K + \Delta U + \Delta Ei$$

Ahora que disponemos de este marco teórico podemos empezar a resolver problemas. Dentro de esas posibilidades puede encontrarse la búsqueda de alguna variable en particular. Si podemos convertir la velocidad en altura, y si podemos calcular una cantidad equivalente de energía mecánica que comparten, entonces podemos encontrar la velocidad que necesitamos para subir a determinada altura y viceversa. Para esto vamos a despejar las variables de cada fórmula y obtener expresiones para cada una:

g= em/m.h
h= em/m.g
m=em/g.h
v= √(2em/m)
m= 2em/v²
ω= √(2em/I)
I= 2em/ω²
x= √(2em/k)
k= 2em/x²

Dónde: **em**=energía mecánica; **g**=gravedad; **h**=altura; **m**= masa; **v**= velocidad traslacional; **ω**= velocidad angular; **I**= momento de inercia; **k**= constante elástica de resorte; **x**=deformación del resorte

c) *La física computacional en la conservación de la energía*

En la actualidad es fundamental integrar las herramientas computacionales a nuestros análisis físicos ya que la cantidad de cálculos, con sus diversos problemas y magnitudes a los que debe enfrentarse una nueva hipótesis, el desarrollo tecnológico, o la especialización en algún área, pueden verse resueltos en una cantidad de tiempo muy reducida.

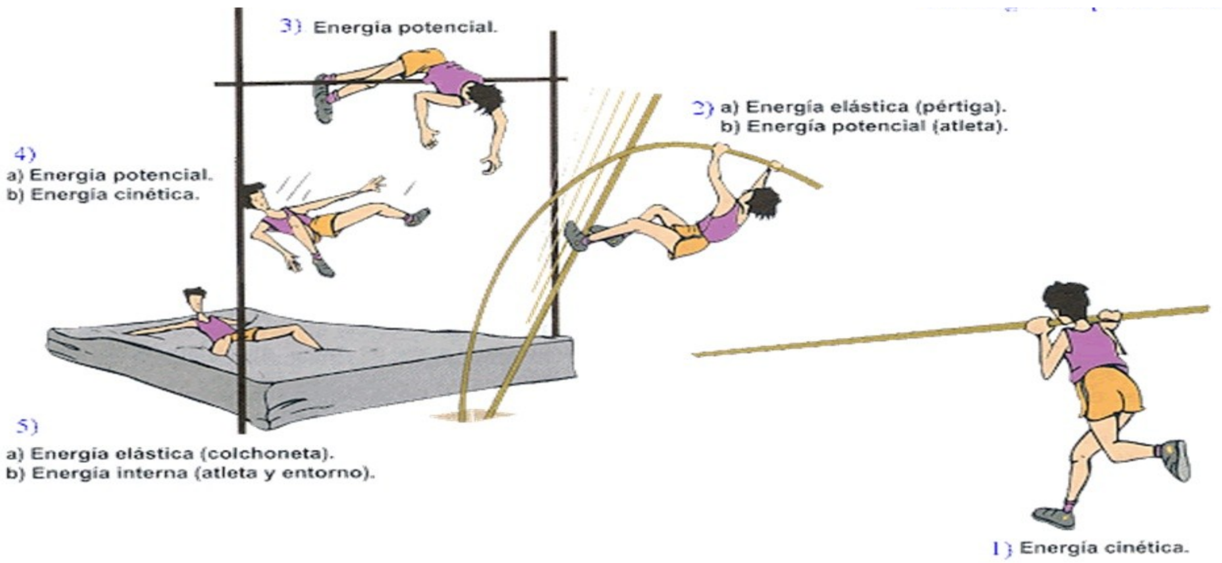
Para esto tenemos un programa hecho en lenguaje Python, que es una herramienta con la que se pueden calcular ordenadamente distintos tipos de energía cinética y potencial, el trabajo de una fuerza externa, la diferencia de energía mecánica y cada variable en particular en alguna etapa del sistema. La creación y uso de un programa en lenguaje Python es de mucha utilidad para llegar a los fines mencionados, por lo que explicaremos de forma detallada con un ejemplo concreto.

Para lo que es la programación en sí se utilizó el entorno “Spyder”, se importaron las librerías correspondientes a las funciones que vamos a usar, se definieron dichas funciones principales que podrían usarse varias veces, luego se armaron los menús y submenús, donde al final de cada operación se vuelve al menú principal. Está pensado para hacer varios cálculos combinando todas las opciones de manera ágil.

El programa tiene una serie de menús y submenús donde se puede seleccionar qué tipo de energía calcular, si potencial o cinética, y dentro de cada opción, cada energía por separado (Ug, Ue, Kt, Kr), sus respectivas cantidades iniciales, finales y la diferencia o suma entre ellas. Luego en el menú principal también podemos calcular la fuerza de rozamiento y su trabajo. También podemos calcular la diferencia de energías mecánicas finales con las iniciales para comprobar la conservación o si el sistema gana o pierde energía. Y por último podemos calcular cada variable por separado a partir de las expresiones generales de cada energía. En cada caso podemos observar graficas que representan las energías calculadas. El programa es bastante sencillo de utilizar pero vamos a explicar su funcionamiento con un ejemplo.

Vamos a aplicar este principio a una situación de deporte olímpico como es el “Salto con garrocha”.

Este deporte es un gran ejemplo de conservación de la energía mecánica y consta de 5 etapas:
1) El atleta comienza a correr convirtiendo la energía interna de su cuerpo en energía cinética.
2) Cuando llega al final del recorrido convierte esa energía cinética acumulada en energía potencial elástica, transfiriéndola a la garrocha, y en energía potencial gravitatoria a medida que va subiendo.
3) En esta etapa toda la energía potencial elástica y cinética se transformó en energía potencial gravitatoria para vencer la fuerza que ejerce el campo gravitatorio terrestre.
4) Una vez que llega a su máxima altura la velocidad se hace cero, y luego empieza a caer, convirtiendo la energía potencial gravitatoria en energía cinética nuevamente.
5) La atleta cae al piso y esa energía cinética se disipa en su cuerpo y en el suelo.



Para simplificar el ejemplo vamos a suponer algunas situaciones:
Suponemos que:

- Los atletas son igual de profesionales y no hay diferencia sustancial en su capacidad técnica.
- La garrocha es la misma para todas las atletas y transfiere toda su energía, por lo que la energía potencial elástica de la garrocha, que funciona como un resorte, no la vamos a considerar en los cálculos. Vamos a pensar que toda la energía cinética se transforma directamente en potencial gravitatoria.
- La pista es la misma para todos por lo que la energía disipada en la fricción con el piso tampoco se toma en cuenta para la etapa que vamos a calcular. Podríamos hacerlo si queremos tener, por ejemplo, información sobre la energía que se pierde en la corrida con la diferencia de masas en distintos atletas.
- La caída no es motivo de este estudio, aunque la podríamos analizar si quisiéramos saber con qué fuerza va a caer, para saber por ejemplo, las protecciones o colchonetas a usar en entrenamientos, o que tipos de caídas entrenar para evitar lesiones.
- Tenemos finalmente un sistema aislado con fuerzas conservativas: “Atleta-Garrocha-Tierra” donde no hay disipación de energía mecánica en la etapa desde que empieza a saltar hasta que empieza a caer.

Actualmente el récord histórico del salto con garrocha lo ostenta Armand Duplantis desde el 25 de febrero del 2022 alcanzando una marca de 6.22m teniendo una masa aproximada de 80 Kg. Supongamos que una persona P de la misma masa que Duplantis y otra H de mayor masa (digamos 85 kg) quieren romper esa marca.

Como venimos mencionando, los factores que hacen obtener energía cinética son la masa y la velocidad, con lo cual podemos calcular la velocidad que necesitaría una persona para superar la altura deseada. Vamos a convertir la velocidad en altura, es decir, la energía cinética en energía potencial.

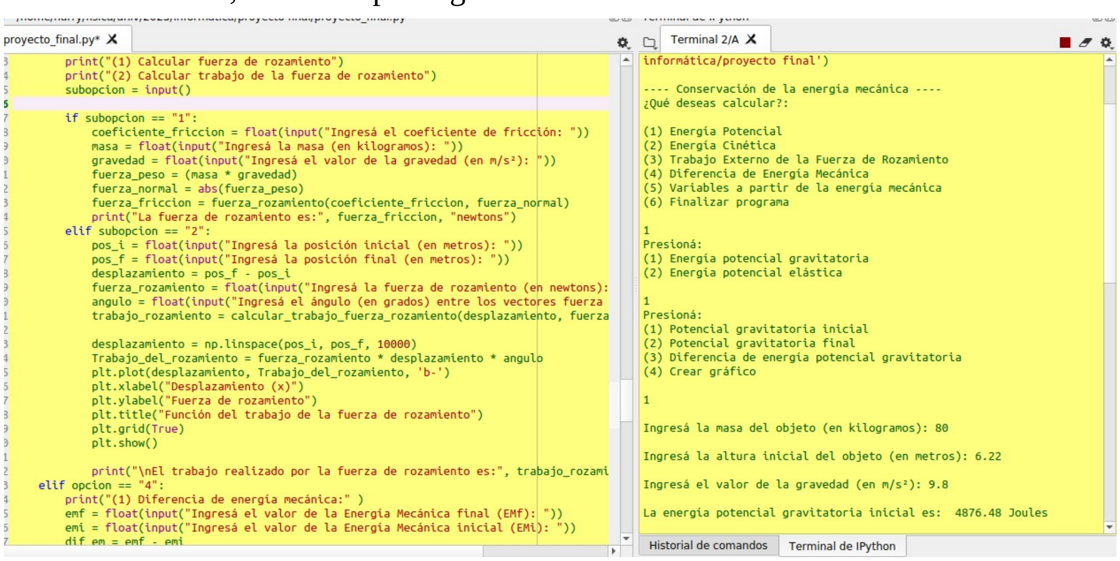
Funcionamiento paso a paso del programa:
“Conservación de la energía mecánica”

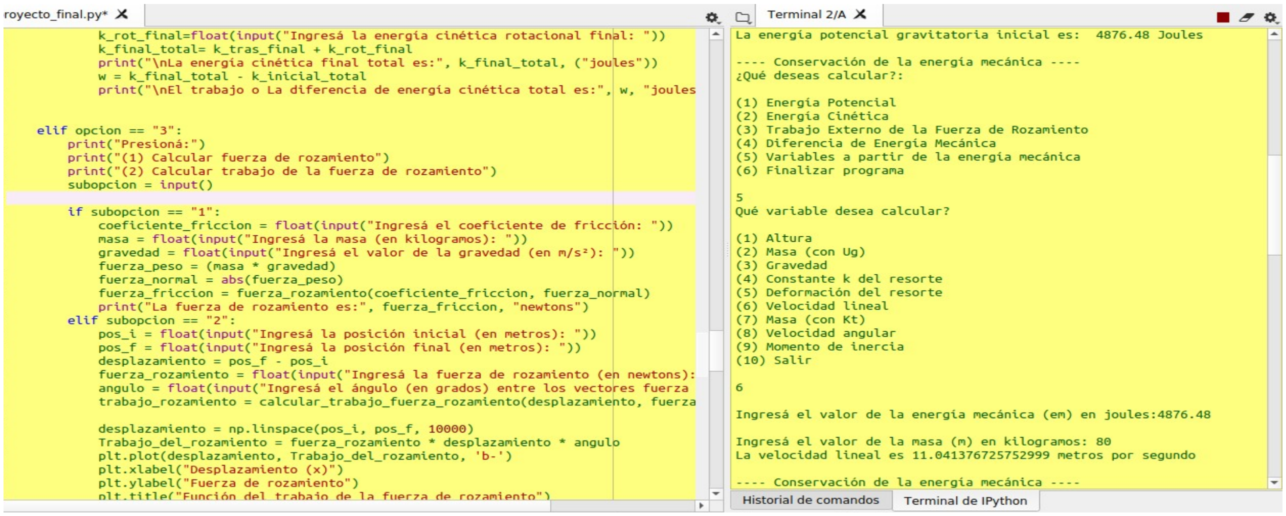
Analicemos entonces el punto de partida de la situación: **Queremos conocer la velocidad que tiene que alcanzar en el momento de saltar, una persona de 80 kg y otra de 85 kg, para alcanzar 6.22m de altura.**

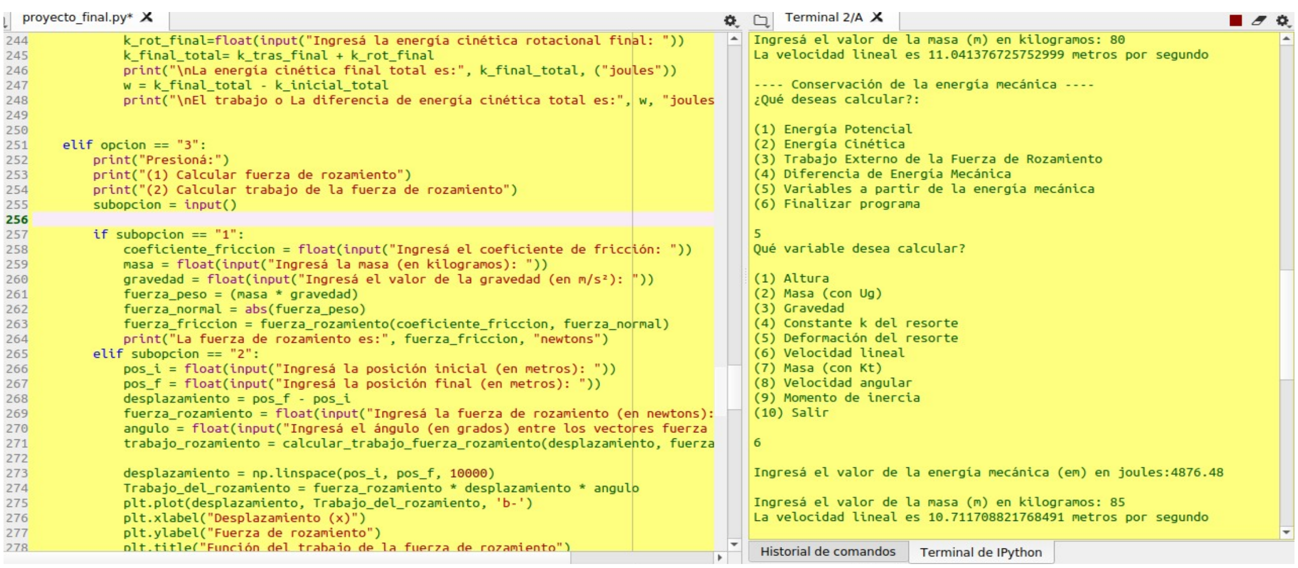
Según nuestro planteo la energía mecánica en ese momento está conformada solamente por la energía potencial gravitatoria ya que la energía cinética en el punto máximo de potencial gravitatoria es 0 (cero) porque no hay movimiento alrededor del centro de masas, ni este se está moviendo y lo mismo sucede con la garrocha que ya transformó toda su potencial elástica. Entonces ya estamos listos para empezar a usar el programa:

Lo primero que hacemos es averiguar cuánta energía mecánica tenemos a esa altura con nuestras atletas.

- Vamos a seleccionar la opción de “Energía potencial”, luego “Energía potencial gravitatoria”. En este caso es indistinto si seleccionamos la “inicial” o la “final”, porque no vamos a compararlas.
- Ingresamos los datos de la masa (80 kg para el primer caso), la altura que queremos alcanzar (6,22 m) y la gravedad (9,8 m/s²). Obtenemos que la energía potencial gravitatoria es de **4876.48 joules.**
- Ahora podemos utilizar esa cantidad de Joules como la energía mecánica que necesitamos convertir. Seleccionamos la opción “(5) Variables a partir de la energía mecánica”, y seleccionamos la opción “Velocidad lineal”. Ingresamos los datos de la energía mecánica que tenemos que reunir (4876.48 joules) y la masa de la atleta (80kg). Obtenemos que el atleta de 80 kg. necesita una velocidad superior a 11.05 metros por segundo para romper el récord. Repetimos el procedimiento para el segundo atleta de 85kg y resulta que necesita superar la velocidad de 10,72 metros por segundo.







4) Conclusión:

Este es un ejemplo sencillo de cómo combinar la física computacional y la ley de la conservación de la energía mecánica en una situación contemporánea. No hace falta mencionar que el programa puede ser mejorado y completado según las especificaciones y necesidades que vayan surgiendo pero es innegable el provecho que podemos obtener de las herramientas informáticas para resolver infinidad de problemas que se nos presenten.

Referencias bibliográficas:

- ***Paul Hewitt, "Física conceptual". Página 117, Capítulo 7: Energía. 10ma edición, 2007***
- ***Serway y Jewett, "Física para ciencias e ingeniería", Volumen 1, 7ma edición, 2008***
- ***http://hyperphysics.phy-astr.gsu.edu/hbasees/work.html#wepr***
- **Jhon Bolaños, "Análisis histórico del principio de conservación de la energía, algunas pautas en la enseñanza de una ley", páginas 25-53 https://bibliotecadigital.univalle.edu.co/bitstream/handle/10893/11323/CB-0525675.pdf?sequence=1&isAllowed=y**
- **César Tomé López** ***en: https://culturacientifica.com/2017/07/04/se-establece-principio-conservacion-la-energia/***
- ***"Trabajo y Energía", presentación dada en 2022 por la cátedra de la materia "Física II" – FCEyN – UNLPam***

Palabras clave:

- ***Python***
- ***Programación***
- ***Energía***
- ***Trabajo***
- ***Física computacional***

Anexo:
Código del programa en lenguaje Python

```
"""Conservación de la energía mecánica"""

#importamos librerías
import math
import numpy as np
import matplotlib.pyplot as plt

#definimos funciones
def potencial_gravitatoria(masa, altura):
    gravedad = float(input("Ingresá el valor de la gravedad (en m/s²): "))
    energia_potencial_grav = masa * gravedad * altura
    return energia_potencial_grav

def cinetica_traslacion(masa, velocidad):
    energia_cinetica_traslacion = (1/2) * masa * (velocidad**2)
    return energia_cinetica_traslacion

def cinetica_rotacion(velocidad_angular, momento_inercia):
    energia_cinetica_rotacion = (1/2) * momento_inercia * (velocidad_angular**2)
    return energia_cinetica_rotacion

def potencial_elastica(constante_resorte, deformacion):
    energia_potencial_elastica = (1/2) * constante_resorte * (deformacion**2)
    return energia_potencial_elastica

def calcular_trabajo_fuerza_rozamiento(desplazamiento, fuerza_rozamiento, angulo, pos_f, pos_i):
    angulo_radianes = math.radians(angulo)
    coseno = math.cos(angulo_radianes)
    desplazamiento = (pos_f - pos_i)
    trabajo_rozamiento = desplazamiento * fuerza_rozamiento * coseno
    return trabajo_rozamiento

#definimos el menú principal
def mostrar_menu():
    print("\n---- Conservación de la energía mecánica ----")
    print("¿Qué deseas calcular?:\n")
    print("(1) Energía Potencial")
    print("(2) Energía Cinética")
    print("(3) Trabajo Externo de la Fuerza de Rozamiento")
    print("(4) Diferencia de Energía Mecánica")
    print("(5) Variables a partir de la energía mecánica")
    print("(6) Finalizar programa")

# Menú y submenús de opciones con bucle
while True:
    mostrar_menu()
    opcion = input()

    if opcion == "1":
        print("Presioná:")
        print("(1) Energía potencial gravitatoria")
        print("(2) Energía potencial elástica")
        subopcion = input()
```

```
if subopcion == "1":
    print("Presioná:")
    print("(1) Potencial gravitatoria inicial")
    print("(2) Potencial gravitatoria final")
    print("(3) Diferencia de energía potencial gravitatoria")
    print("(4) Crear gráfico")
    subopcion_pot_grav = input()

    if subopcion_pot_grav == "1":
        masa = float(input("Ingresá la masa del objeto (en kilogramos): "))
        altura_inicial = float(input("Ingresá la altura inicial del objeto (en metros): "))
        energia_pot_grav_inicial = potencial_gravitatoria(masa, altura_inicial)
        print("\nLa energía potencial gravitatoria inicial es: ", energia_pot_grav_inicial, "Joules")

    elif subopcion_pot_grav == "2":
        masa = float(input("Ingresá la masa del objeto (en kilogramos): "))
        altura_final = float(input("Ingresá la altura final del objeto (en metros): "))
        energia_pot_grav_final = potencial_gravitatoria(masa, altura_final)
        print("\nLa energía potencial gravitatoria final es:", energia_pot_grav_final, "Joules")

    elif subopcion_pot_grav == "3":
        ug_inicial = float(input("Ingresá la energia potencial inicial (en joules): "))
        ug_final= float(input("Ingresá la energia potencial final (en joules): "))
        u_grav = ug_final - ug_inicial
        print("\nLa diferencia de energía potencial gravitatoria es:", u_grav, "joules")

    elif subopcion_pot_grav == "4":
        print("Función Ug: Ug(h) = mgh")
        masa = float(input("Ingresá la masa (en kilogramos): "))
        gravedad = float(input("Ingresá el valor de la gravedad (en m/s²): "))
        altura_inicial = float(input("Ingresá la altura inicial (en metros): "))
        altura_final = float(input("Ingresá la altura final (en metros): "))

        alturas = np.linspace(altura_inicial, altura_final, 10000)
        energias_potenciales = masa * gravedad * alturas

        plt.plot(alturas, energias_potenciales, 'b-')
        plt.xlabel("Altura (h)")
        plt.ylabel("Energía Potencial Gravitatoria (Ug)")
        plt.title("Función de la Energía Potencial Gravitatoria")
        plt.grid(True)
        plt.show()
        valor_maximo = np.max(energias_potenciales)

        print("La energía potencial gravitatoria es: ", valor_maximo, "joules")

elif subopcion == "2":
    print("Presioná:")
    print("(1) Potencial elástica inicial")
    print("(2) Potencial elástica final")
    print("(3) Diferencia de energía potencial elástica")
    print("(4) Crear gráfico")
    subopcion_pot_elas = input()

    if subopcion_pot_elas == "1":
```

```
            constante_resorte = float(input("Ingresá la constante del resorte (en N/m): "))
            deformacion_inicial = float(input("Ingresá la deformación inicial del resorte (en metros):"))
            energia_pot_elas_inicial = potencial_elastica(constante_resorte, deformacion_inicial)
            print("\nLa energía potencial elástica inicial es: ", energia_pot_elas_inicial, "joules")

        elif subopcion_pot_elas == "2":
            constante_resorte = float(input("Ingresá la constante del resorte (en N/m): "))
            deformacion_final = float(input("Ingresá la deformación final del resorte (en metros): "))
            energia_pot_elas_final = potencial_elastica(constante_resorte, deformacion_final)
            print("\nLa energía potencial elástica final es:", energia_pot_elas_final, "joules")

        elif subopcion_pot_elas == "3":
            ue_inicial = float(input("Ingresá la energía potencial elástica inicial (en joules): "))
            ue_final = float(input("Ingresá la energía potencial elástica final (en joules): "))
            u_elas = ue_final - ue_inicial
            print("La diferencia de energía potencial elástica es: ", u_elas, "joules")

        elif subopcion_pot_elas == "4":
            print("Función Ue: Ue(x) = (kx^2) / 2")
            k = float(input("Ingresá el valor de la constante del resorte (en N/m): "))
            x_inicial = float(input("Ingresá el valor de la deformación inicial (en metros): "))
            x_final = float(input("Ingresá el valor de la deformación final (en metros): "))

            def_res = np.linspace(x_inicial, x_final, 10000)
            energias_potenciales = (k * def_res**2) / 2

            plt.plot(def_res, energias_potenciales, 'b-')
            plt.xlabel("Deformación del resorte (x)")
            plt.ylabel("Energía Potencial Elástica (Ue)")
            plt.title("Función de la Energía Potencial Elástica")
            plt.grid(True)
            plt.show()
            valor_maximo = np.max(energias_potenciales)

            print("La energía potencial elástica es: ", valor_maximo, "joules")

elif opcion == "2":
    print("Presioná:")
    print("(1) Cinética de traslación")
    print("(2) Cinética de rotación")
    print("(3) Trabajo o Diferencia de energía cinética total")
    subopcion = input()

    if subopcion == "1":
        print("Presioná:")
        print("(1) Energía cinética traslacional inicial")
        print("(2) Energía cinética traslacional final")
        print("(3) Diferencia de energía cinética traslacional")
        print("(4) Crear gráfico")
        subopcion_k_tras = input()

        if subopcion_k_tras == "1":
            masa = float(input("Ingresá la masa del objeto (en kilogramos): "))
            velocidad_traslacion = float(input("Ingresá la velocidad inicial del objeto (en m/s): "))
            energia_cinetica_traslacion_inicial = cinetica_traslacion(masa, velocidad_traslacion)
            print("\nLa energía cinética de traslación es:", energia_cinetica_traslacion_inicial, "joules")
        elif subopcion_k_tras == "2":
```

```
        masa = float(input("Ingresá la masa del objeto (en kilogramos): "))
        velocidad_traslacion = float(input("Ingresa la velocidad final del objeto (en m/s): "))
        energia_cinetica_traslacion_final = cinetica_traslacion(masa, velocidad_traslacion)
        print("\nLa energía cinética de traslación es:", energia_cinetica_traslacion_final, "joules")
    elif subopcion_k_tras == "3":
        k_tras_inicial = float(input("Ingresá la energía cinética inicial (en joules): "))
        k_tras_final = float(input("Ingresá la energía cinética final (en joules): "))
        dif_k_tras = k_tras_final - k_tras_inicial
        print("\nLa diferencia de energía cinética traslacional es:", dif_k_tras, "joules")
    elif subopcion_k_tras == "4":
        print("Función Kt: Kt(v) = (mv^2) / 2")
        m = float(input("Ingresá el valor de la masa (en metros): "))
        v_inicial = float(input("Ingresá la velocidad inicial (en m/s): "))
        v_final = float(input("Ingresá la velocidad final (en m/s): "))

        dif_v = np.linspace(v_inicial, v_final, 10000)
        energia_k_tras = (m * dif_v**2) / 2

        plt.plot(dif_v, energia_k_tras, 'b-')
        plt.xlabel("Velocidad (v)")
        plt.ylabel("Energía Cinética traslacional (Kt)")
        plt.title("Función de la Energía Cinética traslacional")
        plt.grid(True)
        plt.show()
        valor_maximo = np.max(energia_k_tras)

        print("La energía cinética de traslación es de: ", valor_maximo, "joules")

elif subopcion == "2":
    print("Presioná:")
    print("(1) Energía cinética rotacional inicial")
    print("(2) Energía cinética rotacional final")
    print("(3) Diferencia de energía cinética rotacional")
    print("(4) Crear gráfico")
    subopcion_k_rot = input()

    if subopcion_k_rot == "1":
        momento_inercia = float(input("Ingresá el momento de inercia del objeto (en kg/m²): "))
        velocidad_angular = float(input("Ingresá la velocidad angular inicial del objeto (en rad/s): "))
        energia_cinetica_rotacion = cinetica_rotacion(momento_inercia, velocidad_angular)
        print("\nLa energía cinética de rotación es:", energia_cinetica_rotacion, "joules")
    elif subopcion_k_rot == "2":
        momento_inercia = float(input("Ingresá el momento de inercia del objeto (en k/m²): "))
        velocidad_angular = float(input("Ingresá la velocidad angular final del objeto (en rad/s): "))
        energia_cinetica_rotacion = cinetica_traslacion(momento_inercia, velocidad_angular)
        print("\nLa energía cinética de rotación es:", energia_cinetica_rotacion, "joules")
    elif subopcion_k_rot == "3":
        krot_inicial = float(input("Ingresá la energía cinética rotacional inicial (en joules): "))
        krot_final = float(input("Ingresá la energía cinética rotacional final (en joules): "))
        dif_krot = krot_final - krot_inicial
        print("\nLa diferencia de energía cinética rotacional es:", dif_krot, "joules")
    elif subopcion_k_rot == "4":
        print("Función Kr: Kr(w) = (Iw^2) / 2")
        i = float(input("Ingresá el valor del momento de inercia (en Kg/m²): "))
        w_inicial = float(input("Ingresà la velocidad angular inicial (en rad/s): "))
        w_final = float(input("Ingresá la velocidad angular final rad/s): "))
```

```
            dif_w = np.linspace(w_inicial, w_final, 10000)
            energia_k_rot = (i * dif_w**2) / 2

            plt.plot(dif_w, energia_k_rot, 'b-')
            plt.xlabel("Velocidad angular (w)")
            plt.ylabel("Energía Cinética rotacional (Kr)")
            plt.title("Función de la Energía Cinética rotacional")
            plt.grid(True)
            plt.show()
            valor_maximo = np.max(energia_k_rot)

            print("La energía cinética rotacional es de: ", valor_maximo, "joules")

    elif subopcion == "3":
        k_tras_inicial=float(input("Ingresá la energía cinética traslacional inicial: "))
        k_rot_inicial=float(input("Ingresá la energía cinética rotacional inicial: "))
        k_inicial_total= k_tras_inicial + k_rot_inicial
        print("\nla energía cinética inicial es:", k_inicial_total , "joules")
        k_tras_final=float(input("Ingresá la energía cinética traslacional final: "))
        k_rot_final=float(input("Ingresá la energía cinética rotacional final: "))
        k_final_total= k_tras_final + k_rot_final
        print("\nLa energía cinética final total es:", k_final_total, ("joules"))
        w = k_final_total - k_inicial_total
        print("\nEl trabajo o la diferencia de energía cinética total es:", w, "joules")



elif opcion == "3":
    print("Presioná:")
    print("(1) Calcular fuerza de rozamiento")
    print("(2) Calcular trabajo de la fuerza de rozamiento")
    subopcion = input()

    if subopcion == "1":
        coeficiente_friccion = float(input("Ingresá el coeficiente de fricción: "))
        masa = float(input("Ingresá la masa (en kilogramos): "))
        gravedad = float(input("Ingresá el valor de la gravedad (en m/s²): "))
        fuerza_peso = (masa * gravedad)
        fuerza_normal = abs(fuerza_peso)
        fuerza_friccion = coeficiente_friccion * fuerza_normal
        print("La fuerza de rozamiento es:", fuerza_friccion, "newtons")
    elif subopcion == "2":
        pos_i = float(input("Ingresá la posición inicial (en metros): "))
        pos_f = float(input("Ingresá la posición final (en metros): "))
        desplazamiento = (pos_f - pos_i)
        fuerza_rozamiento = float(input("Ingresá la fuerza de rozamiento (en newtons): "))
        angulo = float(input("Ingresá el ángulo (en grados) entre los vectores fuerza y desplazamiento: "))
        trabajo_rozamiento = calcular_trabajo_fuerza_rozamiento(desplazamiento, fuerza_rozamiento, angulo, pos_i, pos_f)

        desplazamiento = np.linspace(pos_i, pos_f, 10000)
        Trabajo_del_rozamiento = fuerza_rozamiento * desplazamiento * angulo
        plt.plot(desplazamiento, Trabajo_del_rozamiento, 'b-')
        plt.xlabel("Desplazamiento (x)")
        plt.ylabel("Fuerza de rozamiento")
        plt.title("Función del trabajo de la fuerza de rozamiento")
        plt.grid(True)
        plt.show()
```

```
        print("\nEl trabajo realizado por la fuerza de rozamiento es:", trabajo_rozamiento, "joules")
elif opcion == "4":
    print("(1) Diferencia de energía mecánica:" )
    emf = float(input("Ingresá el valor de la Energía Mecánica final (EMf): "))
    emi = float(input("Ingresá el valor de la Energía Mecánica inicial (EMi): "))
    dif_em = emf - emi
    if dif_em == 0:
        print("La diferencia es: ", dif_em)
        print("La energía mecánica se conserva")
    elif dif_em <= 0:
        print("\nLa Energía disipada es: ", dif_em,"joules")
        print("El sistema perdió energía")
    elif dif_em >= 0:
        print("\nLa energía obtenida es de: ", dif_em, "joules")
        print("El sistema ganó energía")

elif opcion == "5":
    print("Qué variable desea calcular?\n")
    print("(1) Altura")
    print("(2) Masa (con Ug)")
    print("(3) Gravedad")
    print("(4) Constante k del resorte")
    print("(5) Deformación del resorte")
    print("(6) Velocidad lineal")
    print("(7) Masa (con Kt)")
    print("(8) Velocidad angular")
    print("(9) Momento de inercia")
    print("(10) Salir ")

    opcion_variables = input()

    if opcion_variables == "1":
        print("Altura (h)")
        em = float(input("Ingresá el valor de la energía mecánica (em) en joules: "))
        g = float(input("Ingresá el valor de la gravedad (g) en m/s²: "))
        m = float(input("Ingresá el valor de la masa (m) en kilogramos: "))
        h = em/(m*g)
        print("La altura es:", h, "metros")

    if opcion_variables == "2":
        print("Masa (m) con Ug")
        em = float(input("Ingresá el valor de la energía mecánica (em) en joules: "))
        g = float(input("Ingresá el valor de la gravedad (g) en m/s²: "))
        h = float(input("Ingresá el valor de la altura en metros: "))
        m = em/(h*g)
        print("La masa es:", m, "kilogramos")

    if opcion_variables == "3":
        print("Gravedad (g)")
        em = float(input("Ingresá el valor de la energía mecánica (em) en joules: "))
        m = float(input("Ingresá el valor de la masa (m) en kilogramos: "))
        h = float(input("Ingresá el valor de la altura en metros: "))
        g = em/(h*m)
        print("La gravedad es:", g, "metros por segundo²")

    if opcion_variables == "4":
        em = float(input("Ingresá el valor de la energía mecánica (em) en joules: "))
```

```
        x = float(input("Ingresá el valor de la deformación (x) en metros: "))
        k = 2*em/(x**2)
        print("La constante k del resorte es: ", k)

    if opcion_variables == "5":
        em = float(input("Ingresá el valor de la energía mecánica (em) en joules: "))
        k = float(input("Ingresá el valor de la constante (k) del resorte: "))
        x = math.sqrt (2*em/k)
        print("La deformación del resorte es", x, "metros")

    if opcion_variables == "6":
        em = float(input("Ingresá el valor de la energía mecánica (em) en joules: "))
        m = m = float(input("Ingresá el valor de la masa (m) en kilogramos: "))
        v = math.sqrt (2*em/m)
        print("La velocidad lineal es", v, "metros por segundo")

    if opcion_variables == "7":
        em = float(input("Ingrese el valor de la energía mecánica (em) en joules: "))
        v = float(input("Ingrese el valor de la velocidad (v) en metros por segundo: "))
        m = 2*em/(v**2)
        print("La masa es", m, "kilogramos")

    if opcion_variables == "8":
        em = float(input("Ingresá el valor de la energía mecánica (em) en joules: "))
        i = i = float(input("Ingresá el valor del momento de inercia (i) en kilogramos por m²: "))
        w = math.sqrt (2*em/i)
        print("La velocidad angular es:", w, "radianes por segundo")

    if opcion_variables == "9":
        em = float(input("Ingresá el valor de la energía mecánica (em) en joules: "))
        w = float(input("Ingresá el valor de la velocidad angular (w) en radianes por segundo: "))
        i =  2*em/(w**2)
        print("El momento de inercia es:", i, "radianes por segundo")

elif opcion == "6":
    break
```