

# ALGORITMO: CUESTIONARIO DE FÍSICA

$E0 = (1,2,3,4,5,6,7)$

GRAVEDAD = 9.81

## Listas

respuestas\_usuario = []

respuestas\_programa = []

## Limpiar lista

funcion limpiar\_lista (respuestas\_usuario, respuestas\_programa):

del respuestas\_usuario [0:3]

del respuestas\_programa [0:3]

## Calificación

funcion calificacion (cont\_f):

calif = (cont\_f\*100/3)

print ("""

Tu calificación final es: """, (" {:.2f} ".format(calif)), "\n")

limpiar\_lista (respuestas\_usuario, respuestas\_programa)

## Correcto o Incorrecto

funcion correcto\_o\_incorrecto (numero\_indice):

contar = 0

```
if (respuestas_usuario[numero_indice]) == (respuestas_programa[numero_indice]):  
    contar = contar + 1  
    print (""  
Correcto"")  
    return contar  
  
else:  
    print (""  
Incorrecto"")  
    return contar
```

### Tema 1\_Operaciones

```
funcion notacion_cientifica(numero_nd):  
    return format(numero_nd, ".1E")
```

```
funcion notacion_decimal(numero_nc):  
    return float(numero_nc)
```

### Tema 1\_Preguntas

#### Notación científica

```
funcion tema1_dificultad1(tema_e, dificultad_e):  
    print ("\nTema 1: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
    respuesta_t1n1_1 = input("")  
¿Para qué se utiliza la notación científica?
```

a) Para expresar cantidades muy grandes o pequeñas

- b) Solo para expresar cantidades muy grandes
  - c) Solo para expresar cantidades muy pequeñas
- """)

```
respuestas_usuario.append(respuesta_t1n1_1)
```

```
respuesta_t1n1_1_p = "a"
```

```
respuestas_programa.append(respuesta_t1n1_1_p)
```

```
cont_1 = correcto_o_incorrecto (0)
```

```
respuesta_t1n1_2 = input("""
```

¿Cuál es la notación científica de 3890000?

favor de expresar el valor como en el siguiente ejemplo:

1400 sería 1.4E+03

→ Solo un decimal

```
""")
```

```
respuestas_usuario.append(respuesta_t1n1_2)
```

```
respuesta_t1n1_2_p =(notacion_cientifica(3890000))
```

```
respuestas_programa.append(respuesta_t1n1_2_p)
```

```
cont_2 = correcto_o_incorrecto (1)
```

```
respuesta_t1n1_3 = float(input("""
```

Para el número en notación científica a notación decimal:

1.7E+05

"""))

```
respuestas_usuario.append(respuesta_t1n1_3)
```

```
respuesta_t1n1_3_p = (notacion_decimal ("1.7E+05"))
```

```
respuestas_programa. append(respuesta_t1n1_3_p)
```

```
cont_3 = correcto_o_incorrecto (2)
```

```
cont_f = cont_1 + cont_2 + cont_3
```

```
calificacion (cont_f)
```

```
funcion tema1_dificultad2(tema_e, dificultad_e):
```

```
print ("\nTema 1: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
respuesta_t1n2_1 = input("""
```

Para convertir un número muy grande a notación científica,  
el punto decimal se moverá hacia la:

a) Derecha

b) Izquierda

""")

```
respuestas_usuario.append(respuesta_t1n2_1)
```

```
respuesta_t1n2_1_p = "b"
```

```
respuestas_programa.append(respuesta_t1n2_1_p)
```

```
cont_1 = correcto_o_incorrecto (0)
```

```
respuesta_t1n2_2 = input("""
```

En la notación científica el número:

- a) Es multiplicado por un número que es elevado a la potencia 10
  - b) Es multiplicado por una potencia base 10 exponente entero
  - c) Es multiplicado por una potencia 10 elevada a un número negativo
- ```
""")
```

```
respuestas_usuario.append(respuesta_t1n2_2)
```

```
respuesta_t1n2_2_p = "b"
```

```
respuestas_programa.append(respuesta_t1n2_2_p)
```

```
cont_2 = correcto_o_incorrecto (1)
```

```
respuesta_t1n2_3 = input("""
```

Convierte el número 0.000793807474 a notación científica:

Favor de expresar la respuesta de la siguiente manera:

a.bE±(exponente)

```
""")
```

```
respuestas_usuario.append(respuesta_t1n2_3)
```

```
respuesta_t1n2_3_p = (notacion_cientifica(0.000793807474))
```

```
respuestas_programa.append(respuesta_t1n2_3_p)
```

```
cont_3 = correcto_o_incorrecto (2)
```

```
cont_f = cont_1 + cont_2 + cont_3
```

```
calificacion (cont_f)
```

```
funcion tema1_dificultad3(tema_e, dificultad_e):
```

```
    print ("\nTema 1: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
    respuesta_t1n3_1 = input("""
```

¿Cuál es la notación científica del resultado de la siguiente operación?

$(123788+98745) / 6748929$

favor de expresar el valor como en el siguiente ejemplo:

a.bE±(exponente)

→ Solo un decimal

```
""")
```

```
respuestas_usuario.append(respuesta_t1n3_1)
```

```
operacion_ad_1 = (123788+98745) / 6748929
```

```
respuesta_t1n3_1_p = (notacion_cientifica (operacion_ad_1))
```

```
respuestas_programa.append(respuesta_t1n3_1_p)
```

```
cont_1 = correcto_o_incorrecto (0)
```

```
respuesta_t1n3_2 = float(input("""
```

Calcula el valor de la siguiente operación y presente su respuesta  
en notación decimal:

$1.7E-05 * 0.134 / 0.001$

```
"""))
```

```
respuestas_usuario.append(respuesta_t1n3_2)
```

```
operacion_ad_2 = 1.7E-05 * 0.134 / 0.001
```

```
respuesta_t1n3_2_p =(operacion_ad_2)
```

```
respuestas_programa.append(respuesta_t1n3_2_p)
```

```
cont_2 = correcto_o_incorrecto (1)
```

```
respuesta_t1n3_3 = input("""
```

Presenta en notación científica el resultado de la siguiente operación:

$23.08/23.89 * (0.0000089 * 3)$

```
""")
```

```
respuestas_usuario.append(respuesta_t1n3_3)
```

```
operacion_ad_3 = 23.08/23.89 * (0.0000089 * 3)
```

```
respuesta_t1n3_3_p = (notacion_cientifica (operacion_ad_3))
```

```
respuestas_programa. append(respuesta_t1n3_3_p)
```

```
cont_3 = correcto_o_incorrecto (2)
```

```
cont_f = cont_1 + cont_2 + cont_3
```

```
calificacion (cont_f)
```

## Tema 2\_Operaciones

### MRU

```
funcion velocidad_mru(distancia_mru, tiempo_mru):
```

```
    velocidad_mru_f1 = distancia_mru/tiempo_mru
```

```
    return (" {:.2f} ".format(velocidad_mru_f1))
```

```
funcion distancia_mru(velocidad_mru, tiempo_mru):
```

```
    velocidad_mru_f2 = velocidad_mru*tiempo_mru
```

```
    return (" {:.2f} ".format(velocidad_mru_f2))
```

```
funcion tiempo_mru(distancia_mru, velocidad_mru):
```

```
    velocidad_mru_f3 = distancia_mru/velocidad_mru
```

```
    return (" {:.2f} ".format(velocidad_mru_f3))
```



## MRUA

"""Tiempo"""

```
funcion tiempo_mrva_vfvia (velocidad_f_mrva, velocidad_i_mrva, aceleracion_mrva):  
    tiempo_mrva_f1 = (velocidad_f_mrva-velocidad_i_mrva)/aceleracion_mrva  
    return "{:.2f}".format(tiempo_mrva_f1))
```

```
funcion tiempo_mrva_vfvid (velocidad_f_mrva, velocidad_i_mrva, distancia_mrva):  
    tiempo_mrva_f2 = distancia_mrva/((velocidad_f_mrva+ velocidad_i_mrva)/2)  
    return "{:.2f}".format(tiempo_mrva_f2))
```

```
funcion tiempo_mrva_da (distancia_mrva, aceleracion_mrva):  
    tiempo_mrva_f3 = ((2* distancia_mrva)/aceleracion_mrva)**(1/2)  
    return "{:.2f}".format(tiempo_mrva_f3))
```

"""Distancia"""

```
funcion distancia_mrva_vita (velocidad_i_mrva, tiempo_mrva, aceleracion_mrva):  
    distancia_mrva_f1 = velocidad_i_mrva*tiempo_mrva +  
    (1/2)*aceleracion_mrva*tiempo_mrva**2  
    return "{:.2f}".format(distancia_mrva_f1))
```

```
funcion distancia_mrva_vfvit (velocidad_f_mrva, velocidad_i_mrva, tiempo_mrva):  
    distancia_mrva_f2 = ((velocidad_f_mrva+velocidad_i_mrva)/2) * tiempo_mrva  
    return "{:.2f}".format(distancia_mrva_f2))
```

```
funcion distancia_mrva_vfvia (velocidad_f_mrva, velocidad_i_mrva,aceleracion_mrva)  
:  
    distancia_mrva_f3 = (velocidad_f_mrva**2-velocidad_i_mrva**2) /  
    (2*aceleracion_mrva)  
    return "{:.2f}".format(distancia_mrva_f3))
```

#### """Velocidad Inicial"""

funcion velocidad\_i\_mrúa\_vfat (velocidad\_f\_mrúa, aceleracion\_mrúa, tiempo\_mrúa):

velocidad\_i\_mrúa\_f1 = velocidad\_f\_mrúa-(aceleracion\_mrúa\*tiempo\_mrúa)

return (" {:.2f} ".format(velocidad\_i\_mrúa\_f1))

funcion velocidad\_i\_mrúa\_vfda (velocidad\_f\_mrúa, distancia\_mrúa, aceleracion\_mrúa)  
:

velocidad\_i\_mrúa\_f2 = (velocidad\_f\_mrúa\*\*2 - 2\*distancia\_mrúa\*aceleracion\_mrúa)  
)\*\*(1/2)

return (" {:.2f} ".format(velocidad\_i\_mrúa\_f2))

#### """Velocidad Final"""

funcion velocidad\_f\_mrúa\_viat (velocidad\_i\_mrúa, aceleracion\_mrúa, tiempo\_mrúa):

velocidad\_f\_mrúa\_f1 = aceleracion\_mrúa\*tiempo\_mrúa + velocidad\_i\_mrúa

return (" {:.2f} ".format(velocidad\_f\_mrúa\_f1))

funcion velocidad\_f\_mrúa\_viad (velocidad\_i\_mrúa, aceleracion\_mrúa, distancia\_mrúa):

velocidad\_f\_mrúa\_f2 = (2\*distancia\_mrúa \* aceleracion\_mrúa +  
velocidad\_i\_mrúa\*\*2)\*\*(1/2)

return (" {:.2f} ".format(velocidad\_f\_mrúa\_f2))

#### """Aceleración"""

funcion aceleracion\_mrúa\_v0vft (velocidad\_f\_mrúa, velocidad\_i\_mrúa, tiempo\_mrúa):

aceleracion\_mrúa\_f1 = (velocidad\_f\_mrúa-velocidad\_i\_mrúa)/tiempo\_mrúa

return (" {:.2f} ".format(aceleracion\_mrúa\_f1))

funcion aceleracion\_mrúa\_v0vfd (velocidad\_f\_mrúa, velocidad\_i\_mrúa,  
distancia\_mrúa)

:

```
    aceleracion_mrua_f2 = (velocidad_f_mrua**2-  
velocidad_i_mrua**2)/(2*distancia_mrua)  
  
    return (" {:.2f} ".format(aceleracion_mrua_f2))
```

## Tema 2\_Preguntas

### MRU y MRUA

funcion tema2\_dificultad1(tema\_e, dificultad\_e):

```
    print ("\nTema 2: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
    respuesta_t2n1_1 = input("""
```

¿Cuál es la fórmula para calcular la velocidad en MRU?

a)  $v = d/t$

b)  $v = d*t$

c)  $v = d+t$

```
""")
```

```
    respuestas_usuario.append(respuesta_t2n1_1)
```

```
    respuesta_t2n1_1_p = "a"
```

```
    respuestas_programa.append(respuesta_t2n1_1_p)
```

```
    cont_1 = correcto_o_incorrecto (0)
```

```
    respuesta_t2n1_2 = float(input("""
```

Calcula la velocidad de un auto que se mueve a 450 m en 120 segundos

(solo ingrese el número con dos decimales)

""))

```
respuestas_usuario.append(str (respuesta_t2n1_2))
```

```
respuesta_t2n1_2_p = velocidad_mru(450.0, 120.0)
```

```
respuestas_programa.append(respuesta_t2n1_2_p)
```

```
cont_2 = correcto_o_incorrecto (1)
```

```
respuesta_t2n1_3 = float(input("""
```

¿Cuál es la aceleración de un objeto que de tener una velocidad de 0 m/s  
pasó a tener una velocidad de 45 m/s en 89 segundos?

(solo ingrese el número con dos decimales)

""))

```
respuestas_usuario.append(str (respuesta_t2n1_3))
```

```
respuesta_t2n1_3_p = aceleracion_mrva_v0vft (45.0, 0.0, 89.0)
```

```
respuestas_programa.append(respuesta_t2n1_3_p)
```

```
cont_3 = correcto_o_incorrecto (2)
```

```
cont_f = cont_1 + cont_2 + cont_3
```

calificacion (cont\_f)

funcion tema2\_dificultad2(tema\_e, dificultad\_e):

```
print ("\nTema 2: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
print ("La misma tematica de nivel uno, pero más difícil")
```

funcion tema2\_dificultad3(tema\_e, dificultad\_e):

```
print ("\nTema 2: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
print ("La misma tematica de nivel uno, pero más difícil que uno y dos")
```

### Tema 3\_Operaciones

#### Caída Libre

""" Altura """

funcion altura\_cl\_t (tiempo\_cl):

```
altura_cl_f1 =(1/2)*GRAVEDAD*tiempo_cl**2
```

```
return (" {:.2f} ".format(altura_cl_f1))
```

funcion altura\_cl\_vit (velocidad\_i\_cl, tiempo\_cl):

```
altura_cl_f2 =(velocidad_i_cl*tiempo_cl)+(1/2)*GRAVEDAD* tiempo_cl**2
```

```
return (" {:.2f} ".format(altura_cl_f2))
```

funcion altura\_cl\_vivft (velocidad\_i\_cl, velocidad\_f\_cl, tiempo\_cl):

```
altura_cl_f3 = ((velocidad_i_cl + velocidad_f_cl)/2)*tiempo_cl
```

```
return (" {:.2f} ".format(altura_cl_f3))
```

""" Tiempo """

funcion tiempo\_cl\_a (altura\_cl):

```
tiempo_cl_f1 = ((2*altura_cl)/GRAVEDAD)**(1/2)
return (" {:.2f} ".format(tiempo_cl_f1))
```

```
funcion tiempo_cl_vivf (velocidad_i_cl, velocidad_f_cl):
    tiempo_cl_f2 = (velocidad_f_cl-velocidad_i_cl)/GRAVEDAD
    return (" {:.2f} ".format(tiempo_cl_f2))
```

```
funcion tiempo_cl_vivfa (velocidad_i_cl, velocidad_f_cl, altura_cl):
    tiempo_cl_f3 = altura_cl/((velocidad_i_cl+velocidad_f_cl)/2)
    return (" {:.2f} ".format(tiempo_cl_f3))
```

""" Velocidad Inicial """

```
funcion velocidad_i_cl_vft (velocidad_f_cl, tiempo_cl):
    velocidad_i_cl_f1 = velocidad_f_cl-GRAVEDAD*tiempo_cl
    return (" {:.2f} ".format(velocidad_i_cl_f1))
```

```
funcion velocidad_i_cl_vfa (velocidad_f_cl, altura_cl):
    velocidad_i_cl_f2 = (velocidad_f_cl**2-2*GRAVEDAD*altura_cl)**(1/2)
    return (" {:.2f} ".format(velocidad_i_cl_f2))
```

```
funcion velocidad_i_cl_at (altura_cl, tiempo_cl):
    velocidad_i_cl_f3 = (altura_cl-(1/2)*GRAVEDAD*tiempo_cl**2)/tiempo_cl
    return (" {:.2f} ".format(velocidad_i_cl_f3))
```

```
funcion velocidad_i_cl_vfat (velocidad_f_cl, altura_cl, tiempo_cl):
    velocidad_i_cl_f4 = ((2*altura_cl)/tiempo_cl)-velocidad_f_cl
    return (" {:.2f} ".format(velocidad_i_cl_f4))
```

""" Velocidad Final """

```
funcion velocidad_f_cl_vit (velocidad_i_cl, tiempo_cl):  
    velocidad_f_cl_f1 = velocidad_i_cl+GRAVEDAD*tiempo_cl  
    return (" {:.2f} ".format(velocidad_f_cl_f1))  
  
funcion velocidad_f_cl_via (velocidad_i_cl, altura_cl):  
    velocidad_f_cl_f2 = (velocidad_i_cl**2+2*GRAVEDAD*altura_cl)**(1/2)  
    return (" {:.2f} ".format(velocidad_f_cl_f2))  
  
funcion velocidad_f_cl_viat (velocidad_i_cl, altura_cl, tiempo_cl):  
    velocidad_f_cl_f3 = ((2*altura_cl)/tiempo_cl)-velocidad_i_cl  
    return (" {:.2f} ".format(velocidad_f_cl_f3))
```

### Tema 3\_Preguntas

#### Caída Libre

```
funcion tema3_dificultad1(tema_e, dificultad_e):  
  
    print ("\nTema 3: " + tema_e + "\n\nNivel " + dificultad_e)  
  
    respuesta_t3n1_1 = input("""  
¿Cuál sería la mejor definición para caída libre?  
  
a) Un objeto que cae  
b) Cualquier objeto bajo la acción de la gravedad en un lugar donde  
    la resistencia del aire es despreciable  
c) Un objeto en caída libre  
""")
```

```
respuesta_t3n1_2 = float(input("""
```

Calcula la altura de un objeto que cae en un tiempo de 67 segundos

[considera que tu velocidad inicial es 0 m/s y tu gravedad de 9.81 m/s<sup>2</sup>]

(solo ingrese el número con dos decimales)

```
"""))
```

```
(altura_cl_t (67.0))
```

```
respuesta_t3n1_3 = float(input("""
```

¿Cuál es la velocidad final de un objeto que recorre 450 m durante 20 segundos y que tiene una velocidad inicial de 13 m/s?

(solo ingrese el número con dos decimales)

```
"""))
```

```
(velocidad_f_cl_viat(13.0, 450.0, 20.0))
```

```
funcion tema3_dificultad2(tema_e, dificultad_e):
```

```
    print ("\nTema 3: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
    print ("La misma temática de nivel uno, pero más difícil")
```

```
funcion tema3_dificultad3(tema_e, dificultad_e):
```

```
    print ("\nTema 3: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
    print ("La misma temática de nivel uno, pero más difícil que uno y dos")
```



## Tema 4\_Operaciones

### Masa y Peso

funcion masa\_mp (peso):

```
masa_mp_f = peso/GRAVEDAD  
return (" {:.2f} ".format(masa_mp_f))
```

funcion peso\_mp (masa):

```
peso_mp_f = masa*GRAVEDAD  
return (" {:.2f} ".format(peso_mp_f))
```

## Tema 4\_Preguntas

### Masa y Peso

funcion tema4\_dificultad1(tema\_e, dificultad\_e):

```
print ("\nTema 4: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
respuesta_t4n1_1 = input("""
```

La unidad de medida de la masa en el SI es:

- a) Gramos
  - b) Onzas
  - c) Kilogramos
- ```
""")
```

```
respuesta_t4n1_2 = float(input("""
```

Calcula la masa de un automóvil que pesa 17500 N

(solo ingrese el número con dos decimales)

```
"""))
```

```
masa_mp (17500.0)
```

```
respuesta_t4n1_3 = float(input("""
```

¿Cuál es el peso de una caja, cuando su masa es igual a 75 Kg?

(solo ingrese el número con dos decimales)

```
"""))
```

```
peso_mp (75.0)
```

```
funcion tema4_dificultad2(tema_e, dificultad_e):
```

```
    print ("\nTema 4: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
    print ("La misma temática de nivel uno, pero más difícil")
```

```
funcion tema4_dificultad3(tema_e, dificultad_e):
```

```
    print ("\nTema 4: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
    print ("La misma temática de nivel uno, pero más difícil que uno y dos")
```

## Tema 5\_Operaciones

### Leyes de Newton

```
funcion fuerza_neta (masa_n, aceleracion_n):
```

```
    fuerza_neta_f = masa_n *aceleracion_n
```

```
    return (" {:.2f}".format(fuerza_neta_f))
```

```
funcion masa_2n (fuerza_n, aceleracion_n):
```

```
masa_2n_f = fuerza_n/acceleracion_n  
return (" {:.2f} ".format(masa_2n_f))
```

```
funcion aceleracion_2n (fuerza_n, masa_n):  
    aceleracion_2n_f = fuerza_n/masa_n  
    return (" {:.2f} ".format(acceleracion_2n_f))
```

## Tema 5\_Preguntas

### Leyes de Newton

```
funcion tema5_dificultad1(tema_e, dificultad_e):  
  
    print ("\nTema 5: " + tema_e + "\n\nNivel " + dificultad_e)  
  
    respuesta_t5n1_1 = input("""  
"A toda acción corresponde una reacción de igual magnitud,  
pero en sentido contrario"  
  
¿De qué ley estamos hablando?  
  
a) 1ra Ley de Newton  
b) 2da Ley de Newton  
c) 3ra Ley de Newton  
  
""")
```

```
    respuesta_t5n1_2 = float(input("""  
¿Cuál es la Fuerza Neta de un objeto con una masa de 40 Kg y una  
aceleración = 3.5 m/s^2?
```

(solo ingrese el número con dos decimales)

```
"""))
```

```
(fuerza_neta (40.0,3.5))
```

```
respuesta_t5n1_3 = input("""
```

¿Qué otro nombre se le da a la primera Ley de Newton?

a) Karma

b) Ley de la Inercia

c) Ley de los objetos estáticos

```
""")
```

```
funcion tema5_dificultad2(tema_e, dificultad_e):
```

```
    print ("\nTema 5: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
    print ("La misma temática de nivel uno, pero más difícil")
```

```
funcion tema5_dificultad3(tema_e, dificultad_e):
```

```
    print ("\nTema 5: " + tema_e + "\n\nNivel " + dificultad_e)
```

```
    print ("La misma temática de nivel uno, pero más difícil que uno y dos")
```

### Tema y Nivel

```
funcion tema_y_nivel ():
```

```
    tema = int(input("""
```

- 1.- Notación científica
- 2.- MRU y MRUA
- 3.- Caída Libre
- 4.- Masa y Peso
- 5.- Leyes de Newton

""))

if tema < 1 or tema > 5:

tema = int(input("""

Lo siento el número seleccionado no está en los parámetros,  
por favor ingrese un número que este dentro de los parámetros:

- 1.- Notación científica
- 2.- MRU y MRUA
- 3.- Caída Libre
- 4.- Masa y Peso
- 5.- Leyes de Newton

""))

dificultad = int(input("""

¿Qué nivel deseas completar? Escribe el número:

- 1.- Sencillo
- 2.- Normal
- 3.- Avanzado

```
"""))
```

```
if dificultad < 0 or dificultad > 3:
```

```
    dificultad = int(input("""
```

Lo siento el número seleccionado no está en los parámetros,  
por favor ingrese un número que este dentro de los parámetros:

1.- Sencillo

2.- Normal

3.- Avanzado

```
"""))
```

```
if tema == 1 and dificultad == 1:
```

```
    tema1_dificultad1("Notación Cinética", "1")
```

```
elif tema == 1 and dificultad == 2:
```

```
    tema1_dificultad2("Notación Cinética", "2")
```

```
elif tema == 1 and dificultad == 3:
```

```
    tema1_dificultad3("Notación Cinética", "3")
```

```
elif tema == 2 and dificultad == 1:
```

```
    tema2_dificultad1("MRU y MRUA", "1")
```

elif tema == 2 and dificultad == 2:

tema2\_dificultad2("MRU y MRUA", "2")

elif tema == 2 and dificultad == 3:

tema2\_dificultad3("MRU y MRUA", "3")

elif tema == 3 and dificultad == 1:

tema3\_dificultad1("Caída Libre", "1")

elif tema == 3 and dificultad == 2:

tema3\_dificultad2("Caída Libre", "2")

elif tema == 3 and dificultad == 3:

tema3\_dificultad3("Caída Libre", "3")

elif tema == 4 and dificultad == 1:

tema4\_dificultad1("Masa y Peso", "1")

elif tema == 4 and dificultad == 2:

tema4\_dificultad2("Masa y Peso", "2")

elif tema == 4 and dificultad == 3:

tema4\_dificultad3("Masa y Peso", "3")

```
elif tema == 5 and dificultad == 1:
```

```
    tema5_dificultad1("Leyes de Newton", "1")
```

```
elif tema == 5 and dificultad == 2:
```

```
    tema5_dificultad2("Leyes de Newton", "2")
```

```
elif tema == 5 and dificultad == 3:
```

```
    tema5_dificultad3("Leyes de Newton", "3")
```

```
##### Algoritmo principal #####
```

```
print ("¡Hola usuario! ¿Qué tema deseas repasar? Por favor ingresa el número")
```

```
tema_y_nivel ()
```

```
continuar_p = input("""
```

```
¿Deseas repasar otro tema?
```

```
a) Sí
```

```
b) No
```

```
""")
```

```
while continuar_p == "a" or continuar_p == "A":
```

```
    tema_y_nivel ()
```

```
    continuar_p = input("""
```

```
¿Deseas repasar otro tema?
```

```
a) Sí
```

```
b) No
```

```
""")
```

```
EF ("¡Gracias por repasar en este programa!")
```