



ISIS-1221

INTRODUCCIÓN A LA PROGRAMACIÓN

Nivel 1 – Proyecto Calculadora Fuerzas

Objetivo general

El objetivo general de este proyecto es que usted practique los conceptos estudiados en el nivel 1 del curso. Recuerde que este proyecto debe realizarse de forma **completamente individual**.

Objetivos específicos

1. Crear funciones.
2. Llamar funciones con parámetros.
3. Llamar funciones desde otras funciones (composición de funciones).
4. Crear y usar un módulo.
5. Probar las funciones de un módulo.
6. Construir interfaces de usuario basadas en consola.

Para lograr lo anterior, en este proyecto se va a crear una aplicación que permite hacer cálculos de fuerzas para el caso de problemas de fricción y movimiento circular vertical.

Actividad 0: Preparación del ambiente de trabajo

1. Cree una carpeta para trabajar, poniéndole su nombre o login.
2. Abra Spyder y cambie la carpeta de trabajo para que sea la carpeta que creó.

Actividad 1: Revisión de la documentación

3. Descargue de Brightspace el archivo “Descripción.pdf” que tiene ayuda importante que le servirá para realizar su proyecto. En particular, presenta la descripción general de la aplicación que usted va a construir y las ecuaciones relevantes de fuerzas que necesita conocer para resolver los problemas.
4. Lea completamente y estudie cuidadosamente ese archivo antes de empezar a trabajar en su programa.

Actividad 2: Construir un módulo de fuerza

5. Usando Spyder, cree en su carpeta de trabajo un nuevo archivo con el nombre “calculadora_fuerzas.py”. En este archivo usted va a construir su módulo en el que va a hacer varios cálculos sobre aceleraciones, masas y fuerzas que se apliquen sobre una caja; y movimiento circular vertical de una pelota. En cada caso le indicaremos qué valores conocemos sobre la caja o pelota y qué queremos calcular.
6. Defina e implemente funciones en su nuevo archivo de acuerdo con la siguiente información.

ATENCIÓN: para asegurar que no haya problemas durante el proceso de calificación, usted debe definir las funciones con los nombres, parámetros y tipos exactos que se presentan a continuación. Las funciones deben estar definidas en el mismo orden.

Nombre de la función	calcular_peso	
Descripción de la función	Calcula el peso de una caja a partir de su masa.	
Parámetros		
Nombre	Tipo	Descripción
masa	float	Cantidad de masa de la caja, en kilogramos
Retorno	float	La magnitud de la fuerza que representa el peso, medida en Newtons ($\text{kg} * \text{m} / \text{s}^2$)

Nombre de la función	calcular_friccion	
Descripción de la función	Calcula la fuerza de fricción sobre una caja que se mueve sobre una superficie, a partir de la masa de la caja y del coeficiente de fricción.	
Parámetros		
Nombre	Tipo	Descripción
masa	float	Cantidad de masa de la caja en kilogramos
coeficiente_friccion	float	Coeficiente de fricción o rozamiento del material de la caja y la superficie
Retorno	float	La magnitud de la fuerza de fricción sobre la caja, medida en Newtons ($\text{kg} * \text{m} / \text{s}^2$)

Nombre de la función	calcular_aceleracion_arrastre	
Descripción de la función	Calcula la aceleración del sistema de acuerdo con la ecuación que se encuentra en el archivo “Descripcion.pdf”, en el cual una caja sobre una superficie está siendo movida por una cuerda. Suponemos que la caja se encuentra en movimiento en el momento en que hacemos el cálculo.	
Parámetros		
Nombre	Tipo	Descripción
masa	float	Cantidad de masa de la caja en kilogramos
coeficiente_friccion	float	Coeficiente de fricción o rozamiento del material de la caja y la superficie
tension	float	Magnitud de la fuerza de tensión sobre la cuerda usada para tirar de la caja, en Newtons
Retorno	float	La aceleración que tiene el sistema de fuerzas, en m/s^2 . Si la aceleración es positiva, significa que el objeto está acelerando. Si es negativa, significa que está desacelerando.

Nombre de la función	calcular_velocidad_superior	
Descripción de la función	Calcula la velocidad de la pelota en la parte superior de la trayectoria circular, de acuerdo con la ecuación que se encuentra en el archivo “Descripcion.pdf”. Sabemos que el sistema se encuentra inicialmente en movimiento.	

Parámetros		
Nombre	Tipo	Descripción
longitud_cuerda	float	Longitud de la cuerda, en metros.
Retorno	float	La velocidad de la pelota, en m/s.

Nombre de la función	calcular_fuerza_centripeta	
Descripción de la función	Calcula la fuerza centrípeta ejercida sobre la pelota en un punto de la trayectoria, de acuerdo con la ecuación que se encuentra en el archivo “Descripcion.pdf”. Sabemos que el sistema se encuentra inicialmente en movimiento.	

Parámetros		
Nombre	Tipo	Descripción
masa	float	Cantidad de masa de la pelota en kilogramos
velocidad	float	Velocidad de la pelota, en m/s
longitud_cuerda	float	Longitud de la cuerda, en metros
Retorno	float	La magnitud de la fuerza centrípeta, en Newtons

Nombre de la función	calcular_tensiones_extremos	
Descripción de la función	Calcula la magnitud de las tensiones en los puntos inferior y superior de la trayectoria circular, de acuerdo con las ecuaciones que se encuentran en el archivo “Descripcion.pdf”. Sabemos que el sistema se encuentra inicialmente en movimiento.	

Parámetros		
Nombre	Tipo	Descripción
masa	float	Cantidad de masa de la pelota en kilogramos
longitud_cuerda	float	Longitud de la cuerda, en metros
Retorno	str	Cadena de la forma “Las tensiones en los puntos inferior y superior de la trayectoria son, respectivamente: X y Z Newtons”, donde X es la tensión inferior, y Z la tensión superior. Las tensiones deben estar redondeadas a dos decimales.

Actividad 3: Construir interfaces de usuario basadas en consola

En esta actividad usted tiene que construir las interfaces basadas en consola para que el usuario interactúe con la aplicación.

ATENCIÓN: las interfaces basadas en consola deben seguir el estándar de construcción de consolas visto en clase.

7. Construya un nuevo archivo Python para cada uno de los 6 problemas que se resuelven con la aplicación. Los nombres deben ser:

- consola_calculo_peso.py
- consola_calculo_friccion.py
- consola_calculo_aceleracion_arrastre.py
- consola_calculo_velocidad_superior.py
- consola_calculo_fuerza_centripeta.py
- consola_calculo_tensiones_extremos.py

ATENCIÓN: estos archivos deben ser creados dentro de la misma carpeta donde se encuentra su módulo de fuerzas.

8. Los nuevos archivos deben importar su módulo de fuerzas para que puedan usar las funciones que definió en el módulo. Por ejemplo, podría usar la siguiente línea para importar el módulo:

```
import calculadora_fuerzas as calc
```

9. Implemente cada uno de los cinco programas de interfaz por consola. Cada uno de estos debe pedirle al usuario los datos necesarios para resolver el problema y debe informarle de su resultado. A modo de ejemplo, la siguiente imagen muestra lo que podría ser el resultado del ejecutar el programa "consola_calculo_peso.py" desde la consola de Windows y desde Spyder.

Desde Spyder:

```
Python 3.7.0 (default, Jun 28 2018, 07:39:16)
Type "copyright", "credits" or "license" for more information.

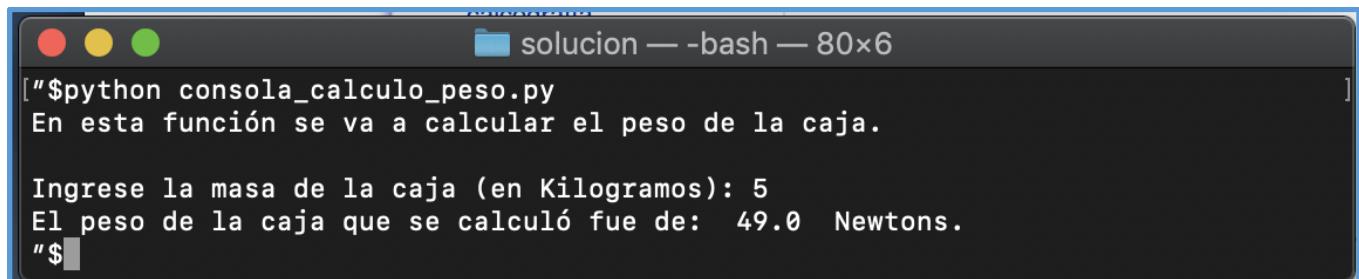
IPython 7.4.0 -- An enhanced Interactive Python.

In [1]: runfile('/Users/nicolascaceres/Desktop/solucion/solucion/consola_calculo_peso.py', wdir='/Users/nicolascaceres/Desktop/solucion/solucion')
En esta función se va a calcular el peso de la caja.

Ingrese la masa de la caja (en Kilogramos): 5
El peso de la caja que se calculó fue de: 49.0 Newtons.

In [2]: |
```

Desde la consola:



```
solucion — bash — 80x6
["$python consola_calculo_peso.py
En esta función se va a calcular el peso de la caja.

Ingrese la masa de la caja (en Kilogramos): 5
El peso de la caja que se calculó fue de: 49.0 Newtons.
"$
```

10. Ejecute cada uno de los 6 programas para asegurar que estén funcionando.

Actividad 4: Verificar el módulo de fuerzas

11. Ejecute cada uno de los 6 programas utilizando los datos que se presentan a continuación y asegúrese que los resultados sean consistentes con lo que se presenta como resultado esperado en la tabla.

Programa	Entradas	Salidas
consola_calculo_peso.py	Masa: 3 Kg	29.4 N
consola_calculo_peso.py	Masa: 9 Kg	88.2 N
consola_calculo_friccion.py	Masa: 5 Kg; Coeficiente: 0.03	1.47 N
consola_calculo_friccion.py	Masa: 8 Kg; Coeficiente: 0.25	19.6 N
consola_calculo_aceleracion_arrastre.py	Masa: 2 Kg; Coeficiente: 0.03; Tensión:15 N	$7.21 \frac{m}{s^2}$
consola_calculo_aceleracion_arrastre.py	Masa: 25 Kg; Coeficiente: 0.05; Tensión:13 N	$0.03 \frac{m}{s^2}$
consola_calculo_velocidad_superior.py	Longitud_cuerda: 5 m	$7.0 \frac{m}{s}$
consola_calculo_velocidad_superior.py	Longitud_cuerda: 15 m	$12.1234 \frac{m}{s}$
consola_calculo_fuerza_centripeta.py	Masa: 8 Kg, Velocidad: 7 m/s, Longitud: 5 m	78.4 N
consola_calculo_fuerza_centridepeta.py	Masa: 12 Kg, Velocidad: 10 m/s, Longitud: 4 m	300.0 N
consola_calculo_tensiones_extremos.py	Masa: 8 Kg, Longitud: 5 m	470.4 N y 0 N
consola_calculo_tensiones_extremos.py	Masa: 12 Kg, Longitud: 4 m	705.6 N y 0 N

12. Si alguno de sus programas presenta algún error o si el resultado es diferente al esperado, revise y corrija su módulo de fuerzas. Es posible que le aparezcan errores causados por errores de tecleo en el nombramiento de las funciones, o errores causados por declarar los parámetros en el orden equivocado. Cada vez que corrija algo, vuelva a realizar las pruebas.

Nota: Las pruebas que está realizando son un mecanismo para identificar posibles problemas con un programa, pero no pueden ser consideradas una garantía de su corrección. Entre más completas estén las pruebas, por ejemplo, verificando casos normales, extremos y anormales, más tranquilidad tendremos de que el programa está bien construido.

Entrega

13. Comprima la carpeta con su proyecto resuelto. El archivo debería llamarse “**N1-PROY-login.zip**”, donde login es su nombre de usuario de Uniandes.
14. Entregue el archivo comprimido a través de Brightspace en la tarea designada como “**Proyecto N1**”.