



ISIS-1221 INTRODUCCIÓN A LA PROGRAMACIÓN

Nivel 2 – Laboratorio condicionales & módulos random y math

Objetivos

1. Conocer y aplicar las funciones de los módulos `random` y `math` de Python. Aprender a buscar, estudiar y entender su documentación.
2. Fomentar la habilidad de descomponer un problema en subproblemas y de implementar funciones que los resuelven, lo que se conoce comúnmente como la técnica de “Dividir y Conquistar”.
3. Practicar el uso de instrucciones condicionales para la solución de problemas por casos.

Módulo `math` y `random` de Python

En este laboratorio se utilizarán dos módulos de Python: `math` y `random`, que implementan funciones que son necesarias para la solución de los ejercicios. La documentación de estos módulos está disponible en la página oficial de Python y se espera que usted sea capaz de buscarla, estudiarla y entenderla.

Recuerde que puede usar la función `help` en la terminal de Spyder para consultar la documentación de una función.

```
Terminal de IPython
Terminal 1/A
Python 3.7.1 (default, Dec 10 2018, 22:54:23) [MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.2.0 -- An enhanced Interactive Python.

In [1]: import random

In [2]: help(random.randint)
Help on method randint in module random:

randint(a, b) method of random.Random instance
    Return random integer in range [a, b], including both end points.
```



ISIS-1221 INTRODUCCIÓN A LA PROGRAMACIÓN

Módulo random

El módulo `random` contiene funciones para la generación de números aleatorios. En este laboratorio se deben utilizar principalmente dos funciones de este módulo:

- `random()`
- `randint(a, b)`

La documentación completa se puede consultar en: <https://docs.python.org/3/library/random.html>

Módulo math

El módulo `math` contiene varias funciones matemáticas. Estas son algunas de ellas:

- `gcd(a,b)`
- `log(x, y)`, `log2(x)`
- `sqrt(x)`
- `sin(x)`, `cos(x)`, `tan(x)`
- `degrees(x)`, `radians(x)`

Este módulo también incluye algunas variables con valores especiales:

- `pi`
- `e`
- `inf`
- `nan`

La documentación completa se puede consultar en: <https://docs.python.org/3/library/math.html>

Actividad 1: Picas y fijas

Picas y fijas es un juego divertido que ayuda a aumentar su nivel de lógica, concentración y análisis. El juego consiste en adivinar un número secreto de 3 dígitos, el cual es generado aleatoriamente por el programa y permanece oculto. El programa suministra ayudas para adivinarlo llamadas PICAS o FIJAS. Se obtiene una FIJA si acierta un dígito y su posición dentro del número secreto. Se obtiene una PICA si acierta un dígito, pero NO acierta su posición dentro del número secreto. Se gana al adivinar el número secreto.



ISIS-1221

INTRODUCCIÓN A LA PROGRAMACIÓN

Ejemplo: El programa crea el número secreto 978, el jugador digita 738, el 8 es una FIJA ya que está en la posición correcta. El 7 es una PICA ya que pertenece al número secreto, pero no coincide en la posición.

Usted debe programar un juego de Picas y Fijas entre un jugador (usuario) y el computador. El computador debe generar aleatoriamente un número de tres dígitos (diferentes) y el jugador debe adivinarlo en 5 turnos **(intentos) máximo**. En cada turno, el jugador ingresa el número de 3 dígitos y el programa le informa la cantidad de fijas y picas que obtuvo. Si el usuario adivina los 3 dígitos (es decir, si llega a tener 3 fijas), el programa debe felicitarlo por haber ganado y ahí termina la ejecución. Si al cabo de 5 intentos el usuario no ha adivinado el número, el programa debe anunciarle que perdió y animarlo a volver a intentarlo. En el transcurso del juego se debe informar al usuario cuántos intentos le quedan aún.

Se le permite imprimir por pantalla el número de 3 dígitos generado aleatoriamente por el programa, para que usted pueda corroborar las picas y fijas que le informa el programa al usuario y verificar que los 3 dígitos del número generado aleatoriamente por el computador son diferentes entre sí. Si los 3 dígitos no son diferentes entre sí, usted debe detener la ejecución del programa (con el botón rojo de parada que se encuentra en la esquina superior derecha de la terminal de IPython) y volver a correrlo.

Ayuda:

* Cree una función que reciba los 6 dígitos (los 3 dígitos que generó el programa y los 3 dígitos que ingresó el usuario) y devuelva el número de picas.

* Cree una función que reciba los 6 dígitos (los 3 dígitos que generó el programa y los 3 dígitos que ingresó el usuario) y devuelva el número de fijas.

Actividad 2: Vectores

En física, un vector es una magnitud física definida en un sistema de referencia que se caracteriza por tener módulo (o longitud), dirección y orientación. En matemáticas se define un vector como un elemento de un espacio vectorial. Los vectores en un espacio euclidiano se pueden representar geoméricamente como segmentos de recta, en el plano bidimensional, o en el espacio tridimensional.

Algunos ejemplos de magnitudes físicas que son magnitudes vectoriales: la velocidad con que se desplaza un móvil, ya que no queda definida tan solo por su módulo que es lo que marca el velocímetro, en el caso de un automóvil, sino que se requiere indicar la dirección (hacia donde se dirige), la fuerza que actúa sobre un objeto, ya que su efecto depende además de su magnitud o módulo, de la dirección en la que actúa; también, el desplazamiento de un objeto, pues es necesario definir el punto inicial y final del movimiento.



ISIS-1221

INTRODUCCIÓN A LA PROGRAMACIÓN

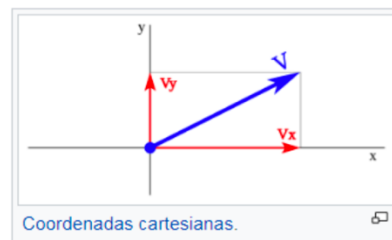
Características de un vector

Un vector se puede definir por sus coordenadas en el plano cartesiano. Si el vector está en el plano xy , se representa de la siguiente manera:

$$\vec{V} = \mathbf{V} = (V_x, V_y)$$

siendo sus coordenadas:

$$V_x, V_y$$



Si consideramos el triángulo formado por los componentes V_x , V_y (como catetos) y V como hipotenusa: se puede calcular V_x multiplicando V por el coseno de α (siendo α el ángulo formado por V_x y V) o multiplicando V por el seno de β (siendo β el ángulo formado por V_y y V). De igual forma, se puede calcular V_y multiplicando V por el seno de α o multiplicando V por el coseno de β .

Módulo de un vector

El módulo de un vector es la longitud del segmento orientado que lo define. Es decir, la distancia que hay del origen al punto definido por sus coordenadas en el plano:

$$|\mathbf{V}| = \sqrt{v_x^2 + v_y^2}$$

Dirección de un vector

La dirección de un vector es la medida del ángulo que hace con una línea horizontal. La siguiente fórmula puede ser usada para encontrar la dirección θ de un vector:

$$\tan \theta = \frac{v_y}{v_x}$$

Por consiguiente:



ISIS-1221 INTRODUCCIÓN A LA PROGRAMACIÓN

$$\theta = \tan^{-1} \frac{v_y}{v_x}$$

Teniendo en cuenta estos conocimientos básicos, construya un programa que:

1. Pida al usuario las coordenadas (x, y) de un vector \mathbf{V} y calcule:
 - a) Las dimensiones de los 3 lados del triángulo rectángulo formado por el vector \mathbf{V} y sus componentes V_x y V_y .
 - b) El área de dicho triángulo.
2. Pida al usuario las coordenadas (x, y) de tres vectores \mathbf{A} , \mathbf{B} y \mathbf{C} (es decir 3 parejas de puntos) y determine:
 - c) Si los tres vectores están alineados, es decir, si apuntan a la misma dirección.
 - d)Cuál de los tres triángulos formados por dichos vectores y sus componentes tiene una mayor área.

Limitaremos el problema a coordenadas positivas. Es decir que todos los puntos se encuentran en el primer cuadrante del plano cartesiano. Antes de empezar a programar, piense qué problemas de los anteriormente planteados pueden ser resueltos por funciones y cómo pueden utilizarse dichas funciones en el programa principal.

Separe su programa en dos archivos: uno para la interfaz del usuario y otro para las funciones que desarrolle.

Nota: Documente sus programas y siga los estándares de buenas practicas de programación que hemos visto en clase.

Actividad 3: Movimiento robótico

Usted quiere anticipar el movimiento de su nuevo robot, que recibió como regalo de cumpleaños. El robot tiene una brújula interna que le permite saber hacia qué punto cardinal está mirando actualmente (Norte, Sur, Este, u Oeste). Además, el robot tiene un control remoto en donde puede voltearse a la izquierda (90°), a la derecha (90°), o dar media vuelta (180°). Dadas estas condiciones, escriba una función que, dadas una orientación inicial y un comando de control remoto de los anteriormente mencionados, pueda predecir la orientación final de su robot.



ISIS-1221

INTRODUCCIÓN A LA PROGRAMACIÓN

Nota: La representación de los puntos cardinales que llegan por parámetro es la siguiente:

- "N", para Norte
- "S", para Sur
- "E", para Este
- "W", para Oeste

La representación de las orientaciones que llegan por parámetro es la siguiente:

- "L", para girar a la izquierda
- "R", para girar a la derecha
- "H", para dar media vuelta
- ".", para mantener la orientación actual

Función requerida

Su solución debe tener una función de acuerdo a la siguiente especificación. Usted puede tener funciones adicionales.

Nombre de la función: movimiento_robot		
Parámetros		
Nombre	Tipo	Descripción
orientacion_actual	str	La orientación actual del robot
giro	str	La acción a ejecutar a partir de la orientación inicial del robot. Debe ser un valor de los siguientes: {"L","H","R","."}
Tipo del retorno		Descripción del retorno
str		La orientación final del robot, debe ser un valor de los siguientes: {"W","N","S","E"}
Funciones y primitivas prohibidas		
Nombre	Descripción	
For	No debería usar la primitiva 'for' para resolver este problema.	
While	No debería usar la primitiva 'while' para resolver este problema.	



ISIS-1221 INTRODUCCIÓN A LA PROGRAMACIÓN

Entrega

Debe entregar un solo archivo comprimido .zip con seis archivos:

- El módulo que contiene las funciones de la actividad 1.
- La interfaz de usuario basada en consola que contiene el programa principal de la actividad 1.
- El módulo que contiene las funciones de la actividad 2.
- La interfaz de usuario basada en consola que contiene el programa principal de la actividad 2.
- El módulo que contiene las funciones de la actividad 3.
- La interfaz de usuario basada en consola que contiene el programa principal de la actividad 3.

Debe entregar un solo archivo comprimido .zip con los archivos que desarrollo para este laboratorio. El nombre del archivo comprimido tiene que seguir la siguiente estructura n2-l2-login1.zip, donde login corresponde a los nombres de usuario de Uniandes. Entregue el archivo a través del BloqueNeon en el laboratorio del nivel 2 designado como “**L2: Condicionales & módulos random y math**”.