

Computación - Tarea 6 (2025-1)

Prof. M en I. Diana Pineda Vázquez
Ayte. Fís. Sebastián Pérez Patricio

31 de octubre de 2024
Fecha de entrega: 7 de noviembre de 2024

1. Elabora un programa en Python que reciba un número de días introducido por el usuario y devuelva el periodo equivalente en años (considerando 365 días), meses (considerando 30 días), semanas (7 días) y días. Inicia creando una lista que contenga la cantidad de días correspondiente a cada intervalo de tiempo. Nota: Recuerda que, en Python, *a módulo b* se escribe como *a%b*.

2. En clase realizamos un programa en Python para hacer el juego del ahorcado con infinitas oportunidades de error. Modificaste el programa para que éste sólo tolerara 5 errores, lanzara el mensaje "Perdiste" y terminara el juego.

Modifica el programa:

- Creando una lista de 10 palabras (de un campo semántico de los siguientes: frutas, plantas, animales, países, ciudades, nombres propios de persona, colores, nombres de científicos, caricaturas, películas. No uses groserías o -- o).
- Usa el método `choice` (viene en las notas) de la librería `random` para que tu programa seleccione una palabra aleatoriamente de la lista creada. Ve este ejemplo que imprime una palabra aleatoria de una lista de frutas:

```
frutas = ['manzana', 'platano', 'naranja', 'fresa', 'pera']
print(random.choice(frutas))
>> fresa
```

- Agregala regla de que si el jugador que adivina repite una letra que yaha sido "descubierta" se cuente como error.
 - Al terminar la partida, dale la opción al jugador de jugar de nuevo: Sí o No y qué le indique al usuario cuántos juegos lleva ganados y cuántos perdidos.
3. Realiza en Python el programa de un juego simple llamado "Adivina mi numero". Se trata de que el usuario adivine, en la menor cantidad de intentos, un número al azar que genera el programa. El número propuesto por el usuario debe ser leído por el programa para compararlo.
 - Genera un número aleatorio en el intervalo 0 a 100, a adivinar.
 - Compara ambos números con `if` mostrando los mensajes correspondientes al usuario, si es menor o mayor el número propuesto al deseado.
 - Para que el usuario tenga varias oportunidades, debe realizar el proceso ciclicamente (`while`). Identifica la condición necesaria.
 - Si el usuario adivina el número de la computadora, i.e. `n == z`, da la instrucción de terminar el ciclo con `break` e indica al usuario que ha adivinado el número.
 - Muestra al usuario en cada intento, el número de intento que lleva.
 - Limita a 5 el número de intentos.