

Introduction to programming-python

Lab Practice 10: Funciones

Programa 1: DNI hackeados

Disponemos de un diccionario de sospechosos. La clave de los elementos del diccionario es un ID sin ningún significado y el valor de cada entrada son los datos de un delincuente en forma de lista. Unos de los campos de la lista es el DNI y se piensa que se ha visto comprometido y la letra ha sido modificada.

Los campos del diccionario son:

- clave--> numerico
- Nombre del delincuente--> cadena
- delito--> cadena
- dni--> cadena
- dirección--> cadena
- teléfono-->numérico (int)

Se pide desarrollar un programa que acceda a todos DNI de todos los delincuentes y recalculé su letra modificándolo en el diccionario.

Para ello, debemos recorrer todo el diccionario y llamar a las funciones correspondientes que nos ayudarán a resolver el problema.

1.- La función SepararNumero(), recibirá como argumento el DNI completo con la letra errónea (en formato cadena ya que está en una sola variable), y nos devolverá el número separado de la letra errónea del DNI. Sólo nos devolverá el número.

2.- Cálculo de la letra del DNI:

Tomamos el número completo de hasta 8 cifras de nuestro DNI, lo dividimos entre 23 y nos quedamos con el resto de dicha división, o dicho de otro modo, calculamos el módulo 23 del DNI.

El resultado anterior es un número entre 0 y 22. A cada uno de estos posibles números le corresponde una letra, según la siguiente tabla:

Resto de la División	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Letra del DIN	T	R	W	A	G	M	Y	F	P	D	X	B	N	J	Z	S	Q	V	H	L	C	K	E

Debemos seguir los siguientes pasos:

1. Generar un bucle que vaya recorriendo uno a uno los elementos del diccionario.
2. Dentro de ese bucle, llamaremos a dos funciones con los argumentos adecuados: SepararNumero(cadena), y CalcularLetra(entero)

3. La función `SepararNumero()`, recibirá como argumento el DNI completo con la letra errónea (en formato cadena ya que está en una sola variable), y nos devolverá el número separado de la letra errónea del DNI. Sólo nos devolverá el número.
4. La función `CalcularLetra()`, recibirá como argumento el número del DNI, y calculará la letra correcta, agregándola al número y devolviendo una cadena con el nuevo DNI al programa principal.
5. Una vez recibido el nuevo DNI, lo modificaremos en el diccionario

DATOS: {1001: ['Alvaro Arrieta', 'Estafador', '51425507Z', 'Lope de Haro 14', 634736248], 1002: ['Rodrigo Cousido', 'Hacker', '63472859N', 'Bernabeu 14', 666394493], 1003: ['Elena Arrieta', 'Timadora', '84572399L', 'Atalayuela 50', 6675849933], 1004: ['Cristina Rueda', 'Chantajista y Timadora', '47558934V', 'Palancas 34', 630768549]}

Programa 2: Siete y Media

Se pide escribir un programa para jugar al juego de cartas de las Siete y Media. El juego consiste en ir dando aleatoriamente una carta de la baraja a un jugador cada vez que lo solicite e ir sumando sus valores hasta llegar a 7.5 puntos o lo más cerca posible. Para ello, los valores de las cartas del 1 al 7 de cada palo puntúan lo mismo que su propio número y las figuras (sota, caballo y rey) valen 0.5 puntos cada una.

La baraja se declara en el programa principal como `int baraja[4][12]` y está representada por una matriz de enteros de 4 filas (representan los palos de la baraja) y 12 columnas (representan cada carta de la baraja). En la matriz una celda tendrá valor 0 si la carta no ha sido repartida y valor 1 si ya ha sido repartida en una mano anterior.

Funcionamiento:

El jugador debe poder jugar tantas partidas como desee, por lo que una vez finalizada la partida, deberemos preguntarle si desea jugar de nuevo, con la consecuente inicialización de todas las variables.

- Para sacar una carta, primero preguntaremos al usuario si desea una nueva carta. En caso afirmativo, llamaremos a la función `NuevaCarta()`, que generará dos valores aleatorios, uno del 1 al 4 (para el palo de la carta), y otro del 1 al 12 para el valor de la carta, generando un valor nuevo si sale el 8 o el 9.
- Si la carta es válida (no ha salido ya), marcaremos la carta en la matriz con un 1, si no es válida, llamaremos de nuevo a la función `NuevaCarta()`
- Por otro lado, llevaremos el control de la suma de las cartas con una variable `suma`. Si la suma excede 7.5, informaremos al jugador de que ha perdido la partida y le preguntaremos si quiere jugar de nuevo.
- Si el jugador se pasa, le informaremos de que ha perdido, y si se planta, sacaremos una carta para la banca. Si la banca se pasa, gana el jugador. A igualdad de puntos, siempre gana la banca.