Documentación programa UNO

**Intervinientes:**

Sergi Torres Cano

Adrián Arias Alonso

Angel Losada Ramos

David Gonzalez Lastra

**Requisitos**:

Realizar un programa en lenguaje **Python** del juego UNO.De manera que un usuario pueda jugar las veces que desee a dicho juego.

**Elementos utilizados:**

En dicho programa utilizamos los siguientes elementos:

1. Menú de información para el usuario

2. Funciones

3. Una variable global

4. Variables.

5. Listas

6. Random

1.Menú de información para el usuario:

Finalidad: Dar información al usuario de las diferentes opciones de las que dispone este juego.

Opciones del menú:

1. Instrucciones: Al pulsar en la primera opción, su finalidad es mostrar las normas que tiene el juego **(UNO),**  dado que puede darse la situación que el usuario sea la primera vez que juega.
2. Jugar: En la segunda opción, colocada, en este lugar dado que ya ha comprendido las opciones de dicho juego.
3. Salir: Una vez que ha finalizado la partida, el programa termina por defecto, pero si el usuario sólo desea ver las instrucciones, debe de poder salir del juego en caso de que no quiera jugar.

Comprobaciones realizadas en el menú:

Se ha creado un **input**  para que el usuario, pueda elegir el número que quiera.Asi como tambien una serie de comprobaciones posteriores al menú:

**mediante if, elif, else:**

-Si la opción es **uno**, debe de ir al menú de instrucciones,

-Si la opción es **dos** accederemos a la **función definirjugadores** en la que el usuario introducirá un número, en el mensaje que le aparecerá un mensaje en el que se indica un mínimo y un máximo permito

-Si por error el usuario introduce un (4) le mostrará un mensaje **Opción incorrecta, vuelva a intentarlo,** por ello si el usuario, pulsa (4) el programa no termina sino que **Se vuelve a ejecutar el menú.**

**Funciones:**

**Carta\_tirada:** destinada a tirar una carta de cada uno de los jugadores participantes en la partida, mediante una lista llamada (**Lista\_jugadores),** una llamada a la función **carta\_puesta**

En esta función tenemos una variable global llamada **turn,** mediante un for se recorre la **Lista de jugadores** en relación al turno, se muestra por pantalla la carta puesta.

En esta función:tenemos una lista llamada **comprobación1** en esta lista lo que consigue es separar mediante un **split** en número de carta, con su color (por ejemplo 2 rojo) lo separará en una lista de 2 valores que serán “2” “rojo”

Mediante otra lista llamada **comprobación2** se hace lo mismo que en la lista **comprobación1** pero con la carta que se encuentra puesta en la mesa.

Si la carta no se puede poner volverá a un input **para solicitar otra carta**, y posteriormente se ha creado, una opción que si el usuario pulsa el número 0 robará una carta

Una vez que el jugador a pulsado **0** se recorre a la lista **mazo**, de la cual se extrae una carta aleatoria que es guardada en una variable llamada **cambio**. Va al jugador que tiene el turno actual, y dentro de su mano le añade la carta que ha robado.. Una vez que el jugador tiene la carta en su mano, esta es eliminada de la lista **mazo** de manera que no pueda repetir la carta.

A continuación se llama a otra función llamada **cambio turno derecha** que posteriormente analizaremos, a la función que llamamos se le pide como parámetros la **lista\_jugadores** y a, **carta\_puesta**(la que se encuentra en la mesa)

Mediante en **elif** se hace comprobación de dos listas(**comprobación1, comprobación2**)teniendo en cuenta que la **comprobación1** es la carta que el jugador tiene en mano, comparada con **comprobación2** que es la carta que esta puesta en mesa.Además con la posición [**0] de ambas listas** se comprueba el número y con la posición **[1]**  se comprueba el color.Si una de las dos condiciones se cumple, podremos tirar la carta.

Volvemos al mazo y actualizamos la carta que acabamos de tirar, y se retira de la mano del jugador que tiene el turno, y posteriormente se comprueba si el jugador tiene cartas, en caso de que no tenga se muestra un mensaje **avisando que ha ganado**, por tanto el programa termina

En el caso de que siga teniendo cartas en mano, se llamará a la función **cambiarTurno Derecha** la que explicaremos más adelante.

**Menú instrucciones:**

Es una función que lo que hace, es llamar a una variable con nombre **reglas** y muestra por pantalla el contenido de la misma.

**Definir\_jugadores;**

Esta es la función que salta a la hora de pulsar la opción 2 en el menú,es el punto en el que tenemos declarada la variable **global turno** contiene un input en el que el jugador introducirá el número de personas que van a jugar, al saltar el input tiene una advertencia que el mínimo de jugadores son 2 y un máximo de 4.

Tenemos una lista llamada **lista de jugadores** que al principio de ejecución de la función se encuentra vacía, con un if se comprueban las normas de límite de jugadores, una vez que sabemos cúantos van a jugar, tenemos que mediante un for declaramos una lista vacía llamada **jugador**

Con la sentencia for(principal), podremos repetir la secuencia, el número de veces como de jugadores tengamos seleccionados . Dentro del for principal tenemos otro secundario en que se ejecutará 7 veces 1 por cada carta que el jugador dispone en su mano.

Las cartas que tiene el usuario son generadas de manera aleatoria, con un random. choice de la lista **mazo** dentro de una variable llamada **agregado,** agregamos de la **lista\_jugadores**  a la mano del jugador con el turno actual. De el mazo debemos de retirar del mazo la carta que se ha guardado en la variable agregado.

Si el jugador, introduce un número que no supere el mínimo o por el contrario supere el máximo, le mostrará un mensaje indicando que el número introducido no cumple los requisitos, Una vez que muestra este mensaje, se vuelve a llamar a la función de definir jugadores, se asigna un turno aleatorio, debido al **random**

**CambioTurnoDerecha :** Recibe como parámetros la lista de jugadores y la carta puesta, recurrimos a la global **turno** teniendo en cuenta que es cambio a la derecha, a la variable turno se le da un valor+1. hacemos una comprobación de manera que el turno este en la lista de jugadores, una vez que se ha cambiado el turno la variable vuelve a tomar el valor de 0 pedimos que nos muestre por pantalla la lista de jugadores y el turno que corresponde y para finalizar volvemos a llamar a la función **carta\_Puesta**