



**Universidad Nacional
Autónoma de México**

Facultad de Ciencias

Adivina el número

Fecha: 04/12/2022

- Alexandra
- Francisco David Valencia Sánchez
- Cesar Eduardo Rosete Gomez

1 Introducción

Para poder demostrar los conocimientos adquiridos en Python, se nos asigno la creación de un programa con tema libre, nuestro proposito en el proyecto es mostrar lo que se puede realizar en unas pocas lineas de codigo, durante el desarrollo de este programa pudimos conocer el comportamiento de los ciclos While, asi mismo como buenos estudiantes autodidactas nos dimos a la tarea de investigar mas allá, esta parte viene mas detallada dentro de la seccion de “Marco teórico”, dentro de este informe vamos a describir el camino para crear un juego el cual consiste en adivinar un número entre el 1 y la cantidad de números x que deseemos.

2 Hipótesis

Nuestro juego consiste en adivinar un número aleatorio, y al cometer un error mostrar un mensaje que nos ayude a encontrarlo, nuestra idea principal es que al ser un juego no seamos tan formales en el lenguaje ni poner instrucciones muy elaboradas, una bienvenida amistosa al usuario es lo primero que vea el jugador, además podemos ofrecer como ya mencionamos una serie de pistas, como por ejemplo, si el jugador nos da un número muy pequeño, decirle que el valor esta muy corto lo que en un mensaje más coloquial seria “Estas frio”, en caso opuesto si el número es demasiado grande mostrar un mensaje coloquial opuesto “Estas muy caliente”, cuando adivine el número acabaría el juego y tendría una muy buena despedida.

3 Marco teórico

Antes de empezar a escribir código hicimos un recuento de nuestros conocimientos que tenemos actualmente, para poder realizar el programa y durante esta investigación nos dimos cuenta de algunas desventajas que teníamos, es claro que vamos ocupar la palabra reservada `print()` para poder mostrar los mensajes que queremos.

Para poder tener el intervalo de números que queremos para escoger podríamos usar el `range()` pero necesitamos que nuestros números sean aleatorios por lo que ocupamos una herramienta muy poderosa que nos ayuda a gener números aleatorios la cual es Random, la tenemos que importar claramente por lo cual usamos `import`, y la función `randit` tambien nos ayudara en el proceso.

Sabíamos que esto se tendría que resolver con un ciclo, después de varios intentos vimos que el ciclo `for` no nos ayudaría para el proyecto que estamos haciendo, por lo que tras una investigación dimos con otro tipo de ciclo llamado `While`.

4 Metodología

Comenzamos definiendo la función: “adivina-el-numero” y tomará un parámetro que llamaremos `x`.

Después le damos la bienvenida al jugador utilizando el comando `print` (“Holiii vamos a jugar”). Mostrando también con el mismo comando cuál es el objetivo del juego. `Print` (“El objetivo es adivinar un número entre el 1 y el `x`”).

Debemos definir el número aleatorio y para eso utilizaremos el módulo `random` (este lo debemos de importar en la primera línea de código) el cual implementa generadores de números pseudoaleatorios para varias distribuciones, utilizando también el módulo `randint` para que nos de un número aleatorio entero. Indicando como límite inferior a 1 y límite superior a `x` en nuestro intervalo. La función `randint` se puede definir de manera formal como: `randint()` es una función incorporada del módulo aleatorio. El módulo aleatorio da acceso a varias funciones útiles y una de ellas es capaz de generar números aleatorios.

número-aleatorio = `random.randint(1, x)`

En la siguiente línea empezamos nuestra base del ciclo indicando que la predicción del usuario es 0 con el objetivo de que no coincida con el número aleatorio obtenido. A la vez que eso se cumple empezamos nuestro ciclo con “`while`” pues las instrucciones se repetirán todas las veces en que el usuario intente adivinar de manera mas formal podemos definir a `While` como:

Los ciclos `while` son una estructura cíclica, que nos permite ejecutar una o varias líneas de código de manera repetitiva sin necesidad de tener un valor inicial e incluso a veces sin siquiera conocer cuando se va a dar el valor final que esperamos.

Con el ciclo `while`, no conoces el cuándo sino el cómo. Es decir, conocer la condición bajo la cual se va a detener el ciclo, pero no sabes cuántas iteraciones tomará eso, ni cuánto tiempo. Por ello se los llama ciclos inde-

terminados.

Iniciamos partiendo de que la predicción sea distinta al número aleatorio y entonces mostramos en la pantalla las instrucciones.

```
Predicción = int (input ("Adivina un número entre el 1 y x "))
```

Utilizamos la función “input” ya que le estamos pidiendo al usuario que ingrese un valor dejando entre llaves el valor de la variable, así mismo usamos el “int” para que trabajemos con enteros, se pone rodeando el input transformando el valor ingresado por el usuario como entero.

Después ponemos nuestros condicionales “if” este será cuando la predicción sea menor que el número aleatorio y mostrará el mensaje

```
Print (" Estas frio chavo")
```

Mientras que el segundo condicional será “elif” que será para el segundo caso en el que la predicción es mayor que el número aleatorio y mostrará el siguiente mensaje.

```
Print ("Estas muy caliente chavo")
```

Cuando el jugador gane se le mostrará el siguiente mensaje

```
Print("¡Felicitaciones! Adivinaste el número número-aleatorio, te ganaste un chocolate")
```

Finalizamos llamando a la función e indicando el valor de nuestro parámetro que en este caso será de 20.

5 Conclusión

Al correr el código y probar nuestro juego los resultados obtenidos fueron los esperados. Utilizamos varios módulos y palabras reservadas aprendidas en el curso, haciendo algunas variaciones en estas, fue como cumplimos con el objetivo principal. Como lo fue al hacer el ciclo necesario de las instrucciones, notamos que era necesario utilizar un ciclo “While” y no un “For” ya que este se repetiría un n número de veces. También hicimos mucho uso del

comando print, ya que uno de nuestros puntos a cumplir era hacer nuestro juego amigable, sencillo y concreto con el usuario, mostrando así instrucciones cortas y divertidas. De la misma manera un módulo clave en el desarrollo de nuestro juego fue “random” la herramienta principal para que python nos diera un número al azar para nuestro juego. Además dentro de este usamos el módulo “randint” resolviendo nuestro problema de que la respuesta ingresada por el usuario se muestre en caracteres y no en un entero que es como lo deseábamos.

Está claro que todos los objetivos fueron cumplidos y los resultados satisfacen lo requerido de nuestro juego utilizando todas las herramientas impartidas en el curso.

6 Referencias

1. Ciclo while en Python. Cómo crear, escribir y usar un while en Python. (2022). Retrieved 4 December 2022, from <https://www.programarya.com/Cursos/Python/Ciclos/Ciclo-while>
2. Función randint() en Python – Acervo Lima. (2022). Retrieved 4 December 2022, from <https://es.acervolima.com/funcion-randint-en-python/>