

---

# Trabajo búsqueda binaria

---

Pablo Zamorano y  
Raúl Moral

---

31/01/2019

---

## Contenido

1- Introducción .....	1
2- Código.....	1
3- Funcionamiento .....	2
4- Aplicaciones búsqueda binaria.....	2
5- Conclusión .....	2
6- Vídeo .....	3

## 1-Introducción

A partir de un código de python, utilizando una combinación binaria hemos conseguido hacer un juego de adivinanzas de números del 1-50

## 2-Código

```
In [4]: import random

intentos=0

print('¡Hola! ¿Cuál es tu nombre?')
nombre = input()

numero=random.randint(1,50)
print('Bueno, '+nombre+', piensa un número entre 1 y 50.')
```

Importamos la librería random, y procedemos a definir la variable intentos. Posteriormente, escribimos lo que corresponderá al saludo de bienvenida al juego, y deberás poner tu nombre. Por último le indicaremos el rango de números con el que jugaremos, y el juego dará inicio.

Ahora utilizaremos un condicional “while” para indicar el número de intentos máximos y que mientras, siga habiendo intentos, escribirá la frase “Venga, tú puedes adivinarlo”.

Después escribiremos el número que creamos que es el correcto, y si el número es mayor pondrá “demasiado grande” y si el número elegido es menor que el de la “adivinanza” dirá “demasiado pequeño”.

La última línea de este condicional dice que en el caso de que adivines el número, el condicional se romperá y saltará a la siguiente parte del código.

```
while intentos<5:
    print ('Venga, tú puedes adivinarlo')
    adivina=input()
    adivina=int(adivina)

    intentos=intentos+1

    if adivina<numero:
        print ('¡Demasiado pequeño!')

    if adivina>numero:
        print('¡Demasiado grande!')

    if adivina==numero:
        break
```

```

if adivina==numero:
    intentos=str(intentos)
    print('Fabuloso, '+nombre+', acertaste el número en '+intentos+' intentos')

if adivina!=numero:
    numero=str(numero)
    print('¡Vaya, '+nombre+'! Yo estaba pensando en el número '+numero)

```

El primero código sirve para, en el caso de que adivines el número, y estés dentro de los intentos, la computadora escribirá la frase estipulada.

En el caso de que no adivines el número dentro de los intentos, la computadora escribirá la frase de derrota

### 3-Funcionamiento

Este trabajo, es un código simple en el que la computadora nos proporcionará un cuadro de texto donde podremos escribir lo que nos pide y el código que utilizará, para el juego.

### 4- Aplicaciones búsqueda binaria

La búsqueda binaria se puede utilizar para acceder a los datos ordenados rápidamente cuando el espacio de memoria es ajustado. Supongamos que desea almacenar un conjunto de 100.000 enteros de 32 bits en una estructura de datos ordenados cuando están disponibles.

- Para construir una máquina virtual donde las etiquetas de casos son enteros individuales. Si tiene 100 casos, puede encontrar la entrada correcta en 6 a 7 pasos usando la búsqueda binaria, donde la secuencia de ramas condicionales toma un promedio de 50 comparaciones.
- Hacer una búsqueda de subcadenas rápida usando matrices de sufijos, que contienen todos los sufijos del conjunto de cadenas de búsqueda en ordenamiento lexicográfico (quería conservar memoria y mantener la implementación simple)
- Encontrar soluciones numéricas a una ecuación, en un menor espacio de tiempo.

### 5- Conclusión

Este trabajo nos parecido sencillo, y de fácil realización. A partir de un código hemos incluido diferentes aspectos, como puede ser la introducción de tu propio nombre y su posterior al uso, en los siguientes diálogos. También nos ha parecido muy divertido.

## 6- Vídeo



2019-01-31-12-14-2  
7.mp4