

Taller 2

Julian Arana^a

^a*Pontificia Universidad Javeriana, Bogota, Colombia*

Abstract

En este algoritmo veremos el desarrollo del algoritmo que permite hacer el juego de adivina el numero donde el usuario lo asigna y el algoritmo lo adivina.

Keywords: algoritmo, escritura formal, adivinar, dividir y vencer.

1. Analisis del problema

En este problema tenemos 2 actores, el software dara un numero aleatorio entre 1 y 1000. El numero que de sera el limite superior que se tendra, por ejemplo si el numero es 500 el rango del numero que el usuario puede poner es de 1 a 500.

Un ejemplo de como deberia funcionar:

|1|2|3|4|5|6|7|8|9|10|

Numero a adivinar: 8.

Dividir el algoritmo a la mitad, si el numero a adivinar es menor al de adivinar, se usara como limite inferior el numero en la mitad, si es al revés, se usa este numero como el limite superior.

|5|6|7|8|9|10| → |8|9|10| → |8|9| → |8|

Numero encontrado.

2. Diseno del problema

El analisis anterior nos permite disenar el problema: definir las entradas y salidas de un posible algoritmo de solucion, que aun no esta definido.

1. *Entradas:* Los limites superior [e] e inferior [b] → [b, e], *El numero a adivinar*.
2. *Salidas:* Un mensaje diciendo si el numero adivinado es mayor, menor o igual al ingresado por el usuario.

*En este documento se realiza el taller 2 de analisis de algoritmos, en el cual se nos pidio elaborar un algoritmo de dividir y vencer para hacer un juego de adivina el numero donde el usuario lo asigna y el algoritmo lo adivina.

Email address: juliana-aranag@javeriana.edu.co (Julian Arana)

3. Algoritmos de solucion

3.1. Algoritmo evidente

Este algoritmo de solucion es una traduccion literal de las deficiones de lo que se quiere resolver.

Algorithm 1 Algoritmo de búsqueda binaria

```

1: procedure ENC_AUX( $b, e, a$ )
2:   if  $b > e$  then
3:     return 0
4:   else
5:      $q \leftarrow \text{round}((b + e)/2)$ 
6:     print ". El número adivinado por el algoritmo es:  $q$ "
7:     if  $q > a$  then
8:       print ". El número es menor"
9:       return ENC_AUX( $b, q, a$ )
10:    else if  $q < a$  then
11:      print ". El número es mayor"
12:      return ENC_AUX( $q, e, a$ )
13:    else
14:      print ". El número es correcto"
15:      return  $q$ 
16:    end if
17:  end if
18: end procedure

```

Algorithm 2 Función para encontrar el número

```

1: procedure ENCONTRAR
2:    $\text{rango} \leftarrow \text{random.randint}(1, 1000)$ 
3:   print ". El rango del número va desde 0 a  $\text{rango}$ "
4:    $\text{numero} \leftarrow \text{int}(\text{input}(\text{"Ingrese el número a adivinar"}))$ 
5:   while  $\text{numero} > \text{rango}$  do
6:      $\text{numero} \leftarrow \text{int}(\text{input}(\text{"El número ingresado es mayor al rango dado. Seleccione otro número"}))$ 
7:   end while
8:    $\text{adivinado} \leftarrow \text{enc\_aux}(0, \text{rango}, \text{numero})$ 
9: end procedure

```
