

## Análisis de Algoritmos 2023

### Grupo 2

Ignacio Alejandro Navarro Oliva y Davor Kissner

Resolución de ejercicios 1, 2, 5, 6 y 7 del TP0 de Repaso de Algoritmia

#### Traza ejercicio 1 de algoritmia

suma	i	j	SALIDA
13	1	3	13
12		2	12
11		1	11
10		0	10
		-1	
23	2	3	23
		2	22
		1	21
		0	20
		-1	
33	3	3	33
32		2	32
		1	31
		0	30
		-1	
43		3	43
42		2	42
41		1	41
40		0	40
		-1	
	5		

#### Traza ejercicio 2

##### metodo

i	vocales	length	vocales[i-1]	vocales[i]	v[i-1]>v[i]	aux
1	[e,u,i,o,a]	5	e	u	false	u
2	[e,i,u,o,a]		u	i	true	
3	[e,i,o,u,a]		u	o	true	
4	[e,i,o,a,u]		u	a	true	
5						

5.  $\log_2(1000000) = 20$ . Se necesitarían como máximo 20 intentos para que se pueda terminar el juego.
6. Se puede hacer una búsqueda secuencial, es decir ir probando número por número hasta encontrar el número  $n$  que necesitamos, en el peor de los casos necesitaremos  $n$  iteraciones, es decir 1 millón en este caso. El otro método es Búsqueda Binaria, si el número está entre 1 y 1 millón, elegimos el número del medio, 500 mil, si nos dice que es mayor, busco el número que se encuentra entre 500 mil y 1 millón y ahora utilizo ese número, si el número es menor, busco el número que está entre 1 y 500 mil y lo utilizo. Esto lo repito la cantidad de veces necesarias hasta encontrar el número. En este caso, si lo hacemos correctamente serían  $\log_2(1000000) = 20$  veces máximo iteraciones.
7. Elegir un pivote diferente en el algoritmo quicksort puede pasar lo siguiente:
  - Último elemento: Puede ser eficiente en muchos casos, pero puede empeorar en listas casi ordenadas (puede llegar a ser peor que bubble sort en dicho caso) .
  - Pivote aleatorio: Suele funcionar bien en promedio y reduce riesgos de mal rendimiento.
  - Pivote mediano de tres (es la media entre el primer elemento, el del medio y el último): Mitiga problemas de peor caso.
  - Pivote fijo (ej: el del medio): Útil si se conoce algo sobre la distribución de valores.