UNIVERSIDAD ICESI

Departamento de TIC
Computación y Estructuras Discretas I
2023-1 - Tarea Integradora 1

Análisis de Complejidad Temporal

Análisis Temporal hashSearch(N)

#	Algoritmo hashSearch(N)		# veces que se ejecuta cada instrucción
1	`if (key == null) {`	C1	1
2	`throw new IllegalArgumentException("Null key");}`	C2	1
3	`int tried = 0;`	C3	1
4	`int j = 0;`	C4	1
5	`Node<V, K> temporal = null;`	C5	1
6	`while (tried != capacity){`	C6	n+1
7	`j = hash(key, tried);`	C7	1
8	`temporal = table.get(j);`	C8	1
9	`if (temporal != null && temporal.getKey().equals(key)){`	C9	n
10	`return temporal.getValue();}`	C10	n
11	`tried += 1;}`	C11	1
12	`return null;`	C12	1
`HashSearch(n)`			`3n + 10`

Análisis Temporal buildMaxHeap(N)

#	Algoritmo hashSearch(N)		# veces que se ejecuta cada instrucción
1	`this.heapSize = array.size()-1;`	C1	1
2	`for (int i = (int)Math.floor(array.size()/2); i>=1; i--){`	C2	n
3	`maxHeapify(i-1);} //Algoritmo Recursivo`	C3	$Log\ n$
`buildMaxHeap(n)`			`n log n`

UNIVERSIDAD ICESI

Departamento de TIC
Computación y Estructuras Discretas I
2023-1 - Tarea Integradora 1

Análisis Temporal heapSort(N)

#	Algoritmo heapSort(N)		# veces que se ejecuta cada instrucción
1	`buildMaxHeap()`	C1	n
2	`for (int i = array.size(); i>=1; i--) {`	C2	n
3	`Collections.swap(array, 0, i - 1);`	C3	1
4	`reduceHeapSize();`	C4	1
5	`maxHeapify(0);}`	C5	Log n
`HashSearch(n)`			`n log n`

Análisis Espacial hashSearch(N)

Tipo	Variable	Tamaño de 1 valor atómico	Cantidad de valores atómicos
Entrada	`Key`	32 bits (Si key == int)	n
Auxiliar	`tried`	32 bits	1
	`J`	32 bits	1
	`temporal`	32 bits (Si value == int)	1
Salida	`null`	Null / 32 bits (Si value == int)	0/1

Complejidad Espacial Total = Entrada + Auxiliar + Salida = n + 4 = θ(n)

Complejidad Espacial Auxiliar = 1 + 1 + 1 = θ(1)

Complejidad Espacial Auxiliar + Salida = 1 + 1 + 1 + 1= θ(1)

Análisis Espacial buildMaxHeap(N)

Tipo	Variable	Tamaño de 1 valor atómico	Cantidad de valores atómicos
Auxiliar	`heapSize`	`32 bits`	`1`
	`array`	`32 bits`	`n`
	`i`	`32 bits`	`1`
Salida	`-`	-	-

Complejidad Espacial Total = Entrada + Auxiliar + Salida = n + 2 = θ(n)

Complejidad Espacial Auxiliar = n + 1 + 1 = θ(n)

Complejidad Espacial Auxiliar + Salida = 0 = θ(1)

Complejidad Espacial Auxiliar + Salida = 0 = θ(1)