Métodos de ordenamiento

Alejandro Giraldo Herrera

Institución universitaria EAM

52410: Ingeniería de software

Ing. Julián Darío Vélez Rodríguez

10 de Marzo de 2023

No. 1	*BUBBLE SORT*	*PASOS*	*Se ejecuta - VECES-*
	`public class BubbleSort {`		
1	`  static int[] sort(int arr[])`	`1 int arr[]`	`1`
2	`  {`	`2 int length = arr.length - 1`	`1`
3 4 5	`    int length = arr.length - 1;`	`3 int i = 0`	`1`
6 7 8	`      for (int i = 0; i < length; i++)`	`4 i < length`	`n + 1`
9	`        for (int j = 0; j < length - i; j++)`	`5 i ++`	`n`
	`          if (arr[j] > arr[j + 1])`	`6 int j = 0`	`n (n)`
10	`          {`	`7 j < length - i`	`n (n + 1 - i)`
11	`            int temp = arr[j];`	`8 j ++`	`n (n - i)`
12	`            arr[j] = arr[j + 1];`	`9 arr[j] > arr[j + 1]`	`n (n - i) (1)`
	`            arr[j + 1] = temp;`	`10 int temp = arr[j]`	`n (n - i) (1)`
13	`          }`	`11 arr[j] = arr[j + 1]`	`n (n - i) (1)`
	`    return arr;`	`12 arr[j + 1] = temp`	`n (n - i) (1)`
	`  }`	`13 return arr`	`1`
	`}`		
Tiempo de ejecución: $T(n) = 5 + 2n + n^2 + n^2 + n - ni + 5n^2 - 5ni \rightarrow T(n) = 5 + 3n + 7n^2 - 6ni$			

No. 2	***INSERTION SORT***	***PASOS***	***Se ejecuta - VECES-***

1
 	
2
3 4
5
 	
6
7
8 9
 	
10
11
 	
12
 	
13
	`public class InsertionSort {`
	`  static int[] sort(int arr[])`
	`  {`
	`    int length = arr.length;`
	`    for (int value = 1; value < length;`
	`++value)`
	`    {`
	`      int key = arr[value];`
	`      int j = value - 1;`
	`      while (j >= 0 && arr[j] > key)`
	`      {`
	`        arr[j + 1] = arr[j];`
	`        j = j - 1;`
	`      }`
	`      arr[j + 1] = key;`
	`    }`
	`    return arr;`
	`  }`
	`}`
		
		1 int arr[]
		2 int length =
		arr.length
		3 int value = 1
		4 value < length
		5 ++ value
		6 int key = arr[value]
		7 int j - value - 1
		8 j >= 0
		9 arr[j] > key
		10 arr[j + 1] = arr[j]
		11 j = j - 1
		12 arr[j + 1] = arr[j]
		13 return arr
			
			1
			1
			1
			n
			n - 1
			(n - 1)(1)
			(n - 1)(1)
			(n - 1)(n + 2)
			(n - 1)(n + 1)
			(n - 1)(n + 1)
			(n - 1)(n + 1)
			(n - 1)(1)
			1

Tiempo de ejecución: $T(n) = 3 + 2n + 3n - 3 + n^2 + 2n - n - 2 + 3n^2 + 3n - 3n - 3$

$\rightarrow T(n) = 4n^2 + 6n - 5$

No. 3	*SELECTION SORT*	*PASOS*	*Se ejecuta - VECES-*
	`public class SelectionSort {`		
1	`  static int[] sort(int arr[])`	1 int arr[]	1
	`  {`	2 int length = arr.length	1
2	`    int length = arr.length;`	3 int value = 0	1
3 4 5	`    for (int value = 0; value < length - 1;`	4 value < length - 1	n
	`value++)`	5 value ++	n - 1
6	`    {`	6 int min_index = value	(n - 1) 1
7 8 9	`      int min_index = value;`	7 j = value	(n - 1) 1
	`      for (int j = value + 1; j < length; j++)`	8 j < length	(n - 1) (n + 1) - value + 1
10	`      {`	9 j ++	(n - 1) (n) - value + 1
	`        if (arr[j] < arr[min_index])`	10 arr[j] < arr[min_index]	(n - 1) (n) - value + 1
11	`        {`	11 min_index = j	(n - 1) (n) - value + 1
	`          min_index = j;`	12 int temp = arr[min_index]	(n - 1) (n) - value + 1
	`        }`	13 arr[min_index] = arr[value]	(n - 1) (n) - value + 1
12	`      }`	14 arr[value] = temp	(n - 1) (n) - value + 1
13	`      int temp = arr[min_index];`	15 return arr	1
14	`      arr[min_index] = arr[value];`		
	`      arr[value] = temp;`		
15	`    }`		
	`    return arr;`		
	`  }`		
	`}`		

Tiempo de ejecución: $T(n) = 3 + 2n + 2n - 2 + n^2 + n - n - 1 - value + 1 + 6n^2 - 6n - 6value + 6$

$\rightarrow T(n) = 7n^2 - 2N - 7value + 7$