Algorithmics	Student information	Date	Number of session
	UO:300896 Surname: De San Claudio Mesa Name: Alejandro	20/02/25	3

Activity 1. [TABLE 1 = BUBBLE ALGORITHM]

n	*t ordered*	*t reverse*	*t random*
10000	310	1429	991
2*10000	1216	5713	3942
2**2*10000	4895	22721	15943
2**3*10000	19877	OoT	OoT
2**4*10000	OoT	OoT	OoT

t_ordered = O(n^2)

t_reverse = O(n^2)

t_random = O(n^2)

Execution times differences happened because of the execution of the "`Vector.interchange(a, j-1, j); //swap"` . Although it is a O(1) method, it execution takes place in each iteration of the t_reverse loop, and some random times in t_random, while it is never executed for t_ordered (it is executed `if (a[j-1] > a[j]))`

Activity 2 [TABLE 2 = SELECTION ALGORITHM]

n	*t ordered*	*t reverse*	*t random*
10000	310	283	307
2*10000	1221	1128	1217
2**2*10000	4898	4517	4885
2**3*10000	19463	18006	19409
2**4*10000	OoT	OoT	OoT

`

Algorithmics	Student information	Date	Number of session
	UO:300896	6/02/25	2
	Surname: De San Claudio Mesa Name: Alejandro		

They are all the same complexity O(n^2) . Also, they are the same complexity as previous methods, but here "Vector.interchange(a, j-1, j); //swap" is executed once per loop (always), instead of inside the nested loop.

Activity 3 [TABLE 3 = INSERTION_ALGORITHM]

n	*t ordered*	*t reverse*	*t random*
10000	LoR	291	148
2*10000	LoR	1137	572
2**2*10000	LoR	4618	2301
2**3*10000	LoR	18235	9207
2**4*10000	LoR	OoT	36770
2**5*10000	LoR	OoT	OoT
2**6*10000	LoR	OoT	OoT
2**7*10000	LoR	OoT	OoT
2**8*10000	50	OoT	OoT
2**9*10000	99	OoT	OoT
2**10*10000	185	OoT	OoT
2**11*10000	373	OoT	OoT
2**12*10000	744	OoT	OoT
2**13*10000	1484	OoT	OoT

They are all O(n) but t_ordered has a massive advantage when it comes to time-efficiency because of the while loop inside of the for loop `while (j >= 0 && pivot < a[j])` only taking place for t_reverse and t_random (in t_reverse it takes the maximum possible iterations).

Activity 4 [TABLE 4 = QUICKSORT ALGORITHM]

Algorithmics	Student information	Date	Number of session
	UO:300896	6/02/25	2
	Surname: De San Claudio Mesa Name: Alejandro		

n	*t ordered*	*t reverse*	*t random*
250000	LoR	LoR	100
2*250000	62	71	199
2**2*250000	127	143	441
2**3*250000	260	291	939
2**4*250000	534	604	2007
2**5*250000	1089	1234	4516
2**6*250000	2244	2554	11073

Activity 5 [TABLE 5 = QUICKSORT + INSERTION ALGORITHM (n=16 M and random))

n	*t random*
Quicksort	23806
Quicksort+Insertion (k=5)	22923
Quicksort+Insertion (k=10)	21349
Quicksort+Insertion (k=20)	23056
Quicksort+Insertion (k=30)	25470
Quicksort+Insertion (k=50)	27663
Quicksort+Insertion (k=100)	25238
Quicksort+Insertion (k=200)	15816
Quicksort+Insertion (k=500)	30837

Algorithmics	Student information	Date	Number of session
	UO:300896	6/02/25	2
	Surname: De San Claudio Mesa Name: Alejandro		

Quicksort+Insertion (k=1000)	40836