Algorithmics	Student information	Date	Number of session
	UO: 301949	27/02/2025	3
	Surname: Sánchez Menéndez		
	Name: Adrián		

Escuela de Ingeniería Informática
Universidad de Oviedo

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Activity 1. Divide and Conquer by subtraction

Subtraction1: O(n)

For this implementation, the size of the problem is not big enough to get a significant result of the increase in the time it takes the algorithm to run. After using repetitions, we can calculate the time for each size and see that it doubles.

Subtraction2: O(n^2)

For this implementation, the size of the problem is increased by 2, and as the theoretical complexity expects, the times are increased by 2^2=4

Subtraction3: O(2^n)

For this implementation, the size of the problem is increased by 1 linearly, and so to compute the time the next iteration will take, we use the expression:

$$t_i = \frac{n_i}{n_j} t_j = \frac{2^i}{2^j} t_j = 2^{i-j} \times t_j$$

which solving for i = j+1, will indicate that the time doubles when the size is increased by 1 and that matches the times obtained by measuring the algorithm.

- Above n=8192, Substraction1 and Substraction2 generate StackOverflowErrors, because the stack gets overloaded with the recursive calls from the method.

- To compute the time for Subtraction3 with n = 80, the formula shown before is needed:

$$t_{80} = 2^{80-25} \times t_{25} = 2^{55} \times 1371\, ms = 4.94 \times 10^{19} ms$$

$$4.94 \times 10^{19} ms \ \times \frac{1\ s}{1000\ ms} \times \frac{1\ min}{60\ s} \times \frac{1\ h}{60\ min} \times \frac{1\ d}{24\ h} \times \frac{1\ y}{365.25\ d}$$

$$= 1{,}565{,}248{,}330\ years$$

- Subtraction4: O(n^3)

n	Substraction4

100	LoR
200	LoR
400	119
800	913
1600	7106
3200	56909
6400	OoT

- Subtraction5: O($3^{n/2}$)

n	Subtraction5
30	457
32	1335
34	4029
36	11662
38	35151
40	OoT

- To compute the time for Subtraction5 with n = 80, the formula shown above is needed:

$$t_{80} = 3^{\frac{80-34}{2}} \times t_{34} = 3^{46} \times 4029\, ms = 3.57 \times 10^{19} ms$$

$$3.57 \times 10^{19} ms \times \frac{1\, s}{1000\, ms} \times \frac{1\, min}{60\, s} \times \frac{1\, h}{60\, min} \times \frac{1\, d}{24\, h} \times \frac{1\, y}{365.25\, d}$$

$$= 1.13 \times 10^{15}\, years$$

Algorithmics	Student information	Date	Number of session
	UO: 301949	27/02/2025	3
	Surname: Sánchez Menéndez		
	Name: Adrián		

Activity 2. Divide and Conquer by division

Division1: O(n)

For this implementation, the size of the problem is increased by 2, and as the theoretical complexity expects, the times are double.

Division2: O(nlog(n))

For this implementation, the size of the problem is increased by 2, and as the theoretical complexity expects, the times are increased by 2^1log2=0.603. In reality, the times are much closer to a complexity of O(n) because n is a very small number and the log(2) is not significant.

Division3: O(n)

For this implementation, the size of the problem is increased by 2, and as the theoretical complexity expects, the times are double.

$$n^{log_b a} = n^{log_2 2} = n$$

- Division4: O(n^2) a<b^k

n	Division4
1000	LoR
2000	LoR
4000	121
8000	483
16000	1834
32000	7301
64000	28912
128000	OoT

- Division5: O(n^2) a>b^k

n	Division5
1000	LoR

2000	113
4000	446
8000	1749
16000	6812
32000	26680
64000	OoT

Activity 3. [TITLE OF THE ACTIVITY]

n	sum1 O(n)	sum2 O(n^{k+1})=O(n)	sum3 O(n)
3	LoR	LoR	LoR
6	LoR	LoR	LoR
12	LoR	LoR	LoR
24	LoR	LoR	91 x10^{-5}
48	LoR	87 x10^{-5}	181 x10^{-5}
96	60 x10^{-5}	172 x10^{-5}	351 x10^{-5}
192	116 x10^{-5}	334 x10^{-5}	698 x10^{-5}
384	226 x10^{-5}	667 x10^{-5}	1425 x10^{-5}
768	439 x10^{-5}	1325 x10^{-5}	2690 x10^{-5}
1536	896 x10^{-5}	2495 x10^{-5}	5229 x10^{-5}
3072	1805 x10^{-5}	4904 x10^{-5}	10764 x10^{-5}
6144	3510 x10^{-5}	9907 x10^{-5}	21564 x10^{-5}
12288	7113 x10^{-5}	StackOverflow	

Algorithmics	Student information	Date	Number of session
	UO: 301949	27/02/2025	3
	Surname: Sánchez Menéndez		
	Name: Adrián		

n	fib1 O(n)	fib2 O(n)	fib3 O(n)	fib4 O(1.6^n)
10	101 x10^{-6}	134 x10^{-6}	202 x10^{-6}	263 x10^{-5}
20	186 x10^{-6}	237 x10^{-6}	360 x10^{-6}	3148 x10^{-4}
30	249 x10^{-6}	343 x10^{-6}	510 x10^{-6}	37.96
40	344 x10^{-6}	487 x10^{-6}	661 x10^{-6}	4655
50	418 x10^{-6}	566 x10^{-6}	831 x10^{-6}	OoT
60	460 x10^{-6}	696 x10^{-6}	972 x10^{-6}	OoT

Activity 4. Petanque championship organization

The algorithm used to solved the complexity of the problem has a complexity of O(n^2), as it is implemented using divide and conquer and the number of recursive calls is 2, the size of the problem is reduced by half in each iteration and the complexity of the code without the recursive calls is O(n^2); so a=2, b=2 and k=2 and as it is a division divide and conquer and $a<b^k$, then the complexity is O(n^k), or O(n^2).

n	t Calendar
2	LoR
4	LoR
8	LoR

Algorithmics	Student information	Date	Number of session
	UO: 301949	27/02/2025	3
	Surname: Sánchez Menéndez		
	Name: Adrián		

16	1.15 x 10^{-2}
32	4 x 10^{-2}
64	0.1344
128	0.5049
256	1.941
512	7.646
1024	31.2
2048	125.3
4096	528.8
8192	2201.9
16384	10533
32768	OoM

OoM = Out of Memory

The size of the problem is increasing by 2 in each iteration, and as the expected complexity of the algorithm predicts, the times increase by 2^2 = 4.