Algorithmics	Student information	Date	Number of session
	UO: UO301879	06/03/2025	3
	Surname: Sariego Sánchez		
	Name: Martín		

Escuela de Ingeniería Informática
Universidad de Oviedo

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Activity 1. Divide and Conquer by subtraction

The time increases make sense considering the theoretical time complexity of each algorithm, but the times are not exactly what would be expected.

The n value where both Subtraction1 and Subtraction2 stop giving times is n= 16384. This is because the implementation uses too much Stack memory and the stack overflows.

The time for n=80 for Subtraction3 would be around 1374921718,4302478 years.

This is calculated by 80-30(last n value measured) = 50 -> t(n=30) * 2^50 -> 38511 * 2^50 = 43359531312416292864. And then converting it to years.

	Subtraction4
n	t (ms)
100	1,41
200	10,57
400	82,51
800	648
1600	4975
3200	39880
6400	OoT

	Subtraction5
n	t(sub5) ms
30	519
32	1278
34	3811
36	11346
38	33718
40	OoT

The time for n=80 for Subtraction5 would be around 11,176.6967years

This is calculated by 80-38(last n value measured) = 42 -> t(n=38) * 3^(42/2) -> 33718 * 3^21 = 3.52702189*10^14. And then converting it to years.

Algorithmics	Student information	Date	Number of session
	UO: UO301879	06/03/2025	3
	Surname: Sariego Sánchez		
	Name: Martín		

Activity 2. Divide and conquer by division

For Division1, the times with lower n values are increasing less than expected, but still make sense given the O(n) complexity.

For Division2 and Division3, with complexities of O(nlogn) and O(n) respectively, the times for all n values do increase as expected.

n	t(div4) ms
1000	4,24
2000	16,89
4000	66,87
8000	269
16000	1065
32000	4309
64000	16917
128000	OoT

n	t(div5) ms
1000	26
2000	103,5
4000	412
8000	1611
16000	6473
32000	26432
64000	OoT

Activity 3. Two basic examples

Table1	VectorSum2		
n	sum1 (ms)	sum2 (ms)	sum3 (ms)
3	0,000044	0,000071	0,000094
6	0,0000668	0,000115	0,000182
12	0,00009	0,000241	0,000378
24	0,0001362	0,000437	0,000774
48	0,000219	0,000835	0,001568
96	0,0004	0,001628	0,003152
192	0,000749	0,003255	0,0067
384	0,001505	0,0067	0,0131
768	0,002974	0,0131	0,0256
1536	0,0061	0,0256	0,052
3072	0,0115	0,0514	0,1043
6144	0,0229	0,103	0,2084
12288	0,0454	OverFlow	0,4118
24576	0,0922		0,82
49152	0,1843		1,66
98304	0,3692		3,36

Algorithmics	Student information	Date	Number of session
	UO: UO301879	06/03/2025	3
	Surname: Sariego Sánchez		
	Name: Martín		

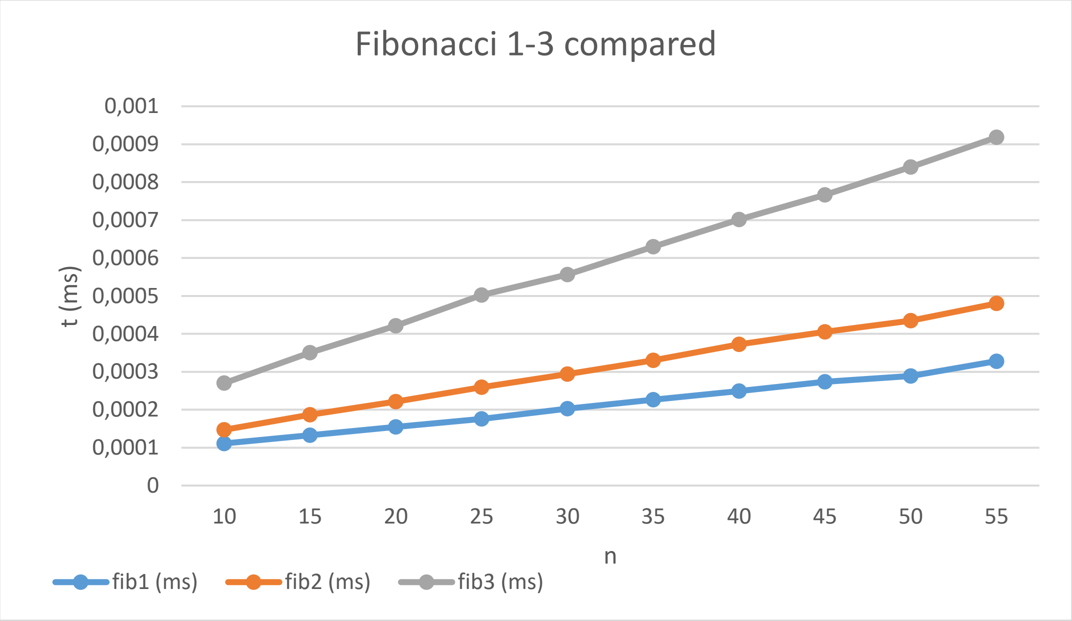
Table2	Fibonacci2			
n	fib1 (ms)	fib2 (ms)	fib3 (ms)	fib4 (ms)
10	0,000089	0,000117	0,000178	0,00221
15	0,000111	0,000147	0,00027	0,02468
20	0,000133	0,000187	0,00035	0,2694
25	0,000155	0,000221	0,000421	3
30	0,000176	0,000259	0,000502	33,64
35	0,000203	0,000294	0,000556	383
40	0,000226	0,00033	0,00063	4096
45	0,000249	0,000372	0,000701	45221
50	0,000274	0,000405	0,000766	OoT
55	0,000289	0,000435	0,00084	
60	0,000328	0,00048	0,000918	

Comparisons			
n	sum1/sum2	sum1/sum3	sum2/sum3
3	0,61971831	0,46808511	0,75531915
6	0,58086957	0,36703297	0,63186813
12	0,37344398	0,23809524	0,63756614
24	0,31167048	0,17596899	0,56459948
48	0,26227545	0,13966837	0,53252551
96	0,24570025	0,12690355	0,51649746
192	0,23010753	0,11179104	0,4858209
384	0,22462687	0,1148855	0,51145038
768	0,2270229	0,11617188	0,51171875
1536	0,23828125	0,11730769	0,49230769
3072	0,22373541	0,11025887	0,4928092
6144	0,2223301	0,10988484	0,49424184
12288		0,11024769	
24576		0,11243902	
49152		0,1110241	
98304		0,10988095	

With this table we see that sum 1 and 2 are pretty similar, but sum1 is still more efficient, and sum3 is the worst out of the three.

Algorithmics	Student information	Date	Number of session
	UO: UO301879	06/03/2025	3
	Surname: Sariego Sánchez		
	Name: Martín		

Comparisons						
n	fib/fib2	fib1/fib3	fib1/fib4	fib2/fib3	fib2/fib4	fib3/fib4
10	0,760683761	0,5	0,040271493	0,657303371	0,052941176	0,080542986
15	0,755102041	0,411111111	0,004497569	0,544444444	0,00595624	0,010940032
20	0,711229947	0,38	0,00049369	0,534285714	0,000694135	0,001299183
25	0,701357466	0,368171021	5,16667E-05	0,524940618	7,36667E-05	0,000140333
30	0,67953668	0,35059761	5,23187E-06	0,515936255	7,69917E-06	1,49227E-05
35	0,69047619	0,365107914	5,30026E-07	0,528776978	7,67624E-07	1,4517E-06
40	0,684848485	0,358730159	5,51758E-08	0,523809524	8,05664E-08	1,53809E-07
45	0,669354839	0,355206847	5,50629E-09	0,530670471	8,22627E-09	1,55016E-08
50	0,67654321	0,35770235		0,528720627		
55	0,664367816	0,344047619		0,517857143		
60	0,683333333	0,357298475		0,522875817		



From this table and graph we can see that Fibonacci1 is the most efficient, and Fibonacci3 the second least effective after Fibonacci4, which is exponential, so it doesn't even fit in the graph.

Activity 4. Petanque championship organization

[ANSWER].