


Algorithmics	Student information	Date	Number of session
	UO:300599	28/01/2025	1
	Surname: Alvarez Fernandez	 Escuela de Ingeniería Informática Universidad de Oviedo	
	Name: Ruben		



## FACTOR 1: PROBLEM SIZE

<code>n = 10000 *** time = 1555 milliseconds</code>
<code>n = 20000 *** time = 6248 milliseconds</code>
<code>n = 40000 *** time = 25388 milliseconds</code>
<code>n = 80000 *** time = Out</code>
<code>n = 160000 *** time = Out</code>
<code>n = 320000 *** time = Out</code>
<code>n = 640000 *** time = Out</code>

## FACTOR 2: COMPUTER PERFORMANCE

### TEST PC\_1

CPU	Intel(R) Core(TM) i5-7500 CPU (4 CPU's) 3.4 GHz
RAM	8192 MB

<code>n = 10000 *** time = 2509 milliseconds</code>
<code>n = 20000 *** time = 10090 milliseconds</code>
<code>n = 40000 *** time = 41910 milliseconds</code>

Algorithmics	Student information	Date	Number of session
	UO:300599	28/01/2025	1
	Surname: Alvarez Fernandez		
	Name: Ruben		

<code>n = 80000 *** time = 0ot</code>
<code>n = 160000 *** time = 0ot</code>
<code>n = 320000 *** time = 0ot</code>
<code>n = 640000 *** time = 0ot</code>

## TEST PC\_2

CPU	Intel Core i5-12400
RAM	16 GB

<code>n = 10000 *** time = 1417 milliseconds</code>
<code>n = 20000 *** time = 6012 milliseconds</code>
<code>n = 40000 *** time = 25309 milliseconds</code>
<code>n = 80000 *** time = 0ot</code>
<code>n = 160000 *** time = 0ot</code>
<code>n = 320000 *** time = 0ot</code>
<code>n = 640000 *** time = 0ot</code>

Algorithmics	Student information	Date	Number of session
	UO:300599	28/01/2025	1
	Surname: Alvarez Fernandez		
	Name: Ruben		

## FACTOR 3: IMPLEMENTATION ENVIROMENT

### JavaA1.java

n = 10000, ***, time = 175 , milliseconds
n = 20000, ***, time = 449 , milliseconds
n = 40000, ***, time = 1750 , milliseconds
n = 80000, ***, time = 6846 , milliseconds
n = 160000, ***, time = 27254 , milliseconds
n = 320000 *** time = Oot
n = 640000 *** time = Oot

Algorithmics	Student information	Date	Number of session
	UO:300599	28/01/2025	1
	Surname: Alvarez Fernandez		
	Name: Ruben		

## FACTOR 4: ALGORITHM THAT IS USED

### PYTHON

V1	V2	V3
n = 10000 *** time = 2539 milliseconds	n = 10000 *** time = 306 milliseconds	n = 10000 *** time = 152 milliseconds

n = 20000 *** time = 1028 9 milliseconds	n = 20000 *** time = 1097 milliseconds	n = 20000 *** time = 550 milliseconds
n = 40000 *** time = 3589 1milliseconds	n = 40000 *** time = 4188 milliseconds	n = 40000 *** time = 2060 milliseconds
n = 80000 *** time = Oot	n = 80000 *** time = 15493 milliseconds	n = 80000 *** time = 7916 milliseconds
n = 160000 *** time = Oot	n = 160000 *** time = 5716 5 milliseconds	n = 160000 *** time = 291 93 millisecond
n = 320000 *** time = Oot	n = 320000 *** time = Oot	n = 320000 *** time = Oot
n = 640000 *** time = Oot	n = 640000 *** time = Oot	n = 640000 *** time = Oot

Algorithmics	Student information	Date	Number of session
	UO:300599	28/01/2025	1
	Surname: Alvarez Fernandez		
	Name: Ruben		

## JAVA WITHOUT OPTIMIZATION

V1	V2	V3
n = 10000, ***, time = 116 , milliseconds	n = 10000, ***, time = 16 , millisecond	n = 10000, ***, time = 9 , milliseconds

n = 20000, ***, time = 439 , milliseconds	n = 20000, ***, time = 48 , milliseconds	n = 20000, ***, time = 26 , milliseconds
n = 40000, ***, time = 1760 , milliseconds	n = 40000, ***, time = 175 , milliseconds	n = 40000, ***, time = 98 , milliseconds
n = 80000, ***, time = 7051 , milliseconds	n = 80000, ***, time = 674 , milliseconds	n = 80000, ***, time = 336 , milliseconds
n = 160000, ***, time = 27872 , milliseconds	n = 160000, ***, time = 2622 , milliseconds	n = 160000, ***, time = 1213 , milliseconds
n = 320000 *** time = Oot	n = 320000, ***, time = 10144 , milliseconds	n = 320000, ***, time = 4567 , milliseconds
n = 640000 *** time = Oot	n = 640000, ***, time = 35936 , milliseconds	n = 640000, ***, time = 17368 , milliseconds

Algorithmics	Student information	Date	Number of session
	UO:300599	28/01/2025	1
	Surname: Alvarez Fernandez		
	Name: Ruben		

## JAVA WITH OPTIMIZATION

V1	V2	V3
n = 10000, ***, time = 117 , milliseconds	n = 10000, ***, time = 16 , milliseconds	n = 10000, ***, time = 9 , milliseconds

n = 20000, ***, time = 444 , milliseconds	n = 20000, ***, time = 47 , milliseconds	n = 20000, ***, time = 27 , milliseconds
n = 40000, ***, time = 1762 , milliseconds	n = 40000, ***, time = 172 , milliseconds	n = 40000, ***, time = 95 , milliseconds
n = 80000, ***, time = 6945 , milliseconds	n = 80000, ***, time = 683 , milliseconds	n = 80000, ***, time = 336 , milliseconds
n = 160000, ***, time = 27790 , milliseconds	n = 160000, ***, time = 2619 , milliseconds	n = 160000, ***, time = 1214 , milliseconds
n = 320000 *** time = Oot	n = 320000, ***, time = 10014 , milliseconds	n = 320000, ***, time = 4588 , milliseconds
n = 640000 *** time = Oot	n = 640000, ***, time = 34891 , milliseconds	n = 640000, ***, time = 17706 , milliseconds

Algorithmics	Student information	Date	Number of session
	UO:300599	28/01/2025	1
	Surname: Alvarez Fernandez		
	Name: Ruben		

#### Conclusion:

In both Python and Java implementations, we can confirm that V2 and V3 optimize execution time. V2 achieves this by modifying the function primoA1 to check if a number is divisible by any other number from 2 to itself, stopping immediately when the condition is met. V3 further optimizes the function by limiting the divisor range from 2 to  $n // 2 + 1$ , thereby reducing the number of iterations.

Comparing the results obtained in Java and Python, we can confirm that compiled languages like Java execute these algorithms faster than interpreted languages like Python.

Finally, we can conclude that compiler optimizations in Java contribute to improved execution performance, further enhancing efficiency.