

ANÁLISIS DE ALGORITMOS

Alumno: Martín Eduardo Barriga Vargas



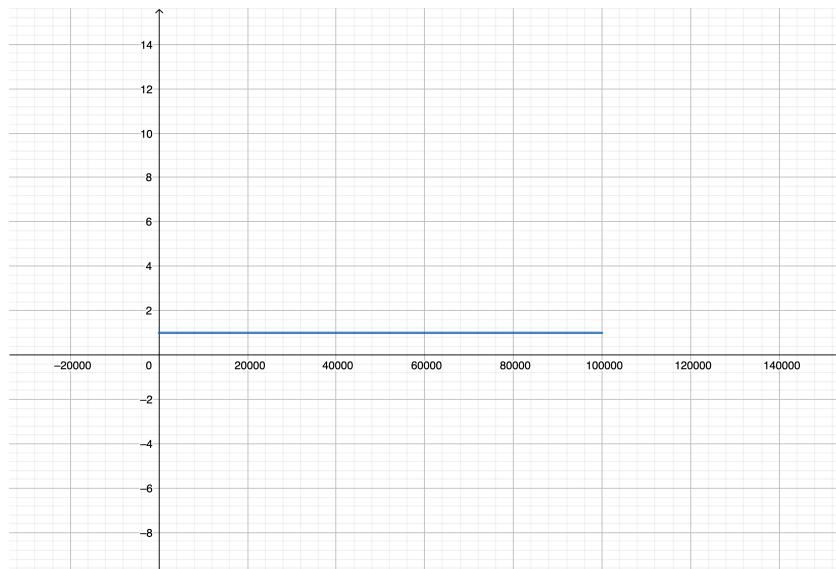
Grupo 3CM3

Ejercicio 03: Graficación de Ordenes
de complejidad

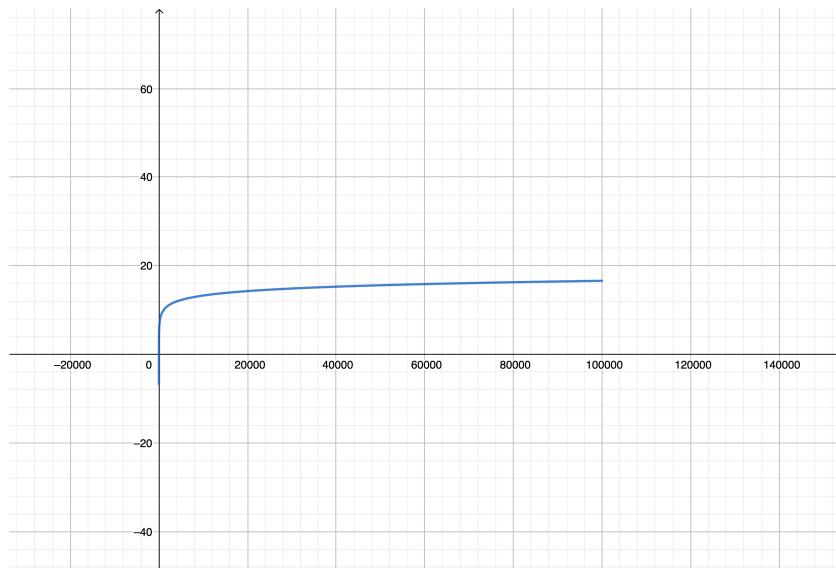
29 de marzo de 2019

1.- Dados los ordenes de complejidad graficar cada uno de estos de manera separada para un rango de $0 < n < 100,000$.

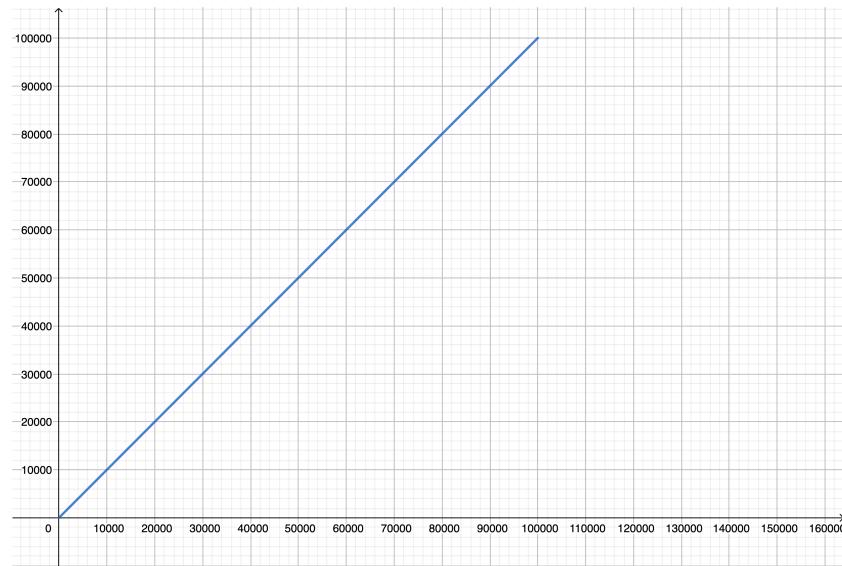
- $O(1)$ Complejidad constante



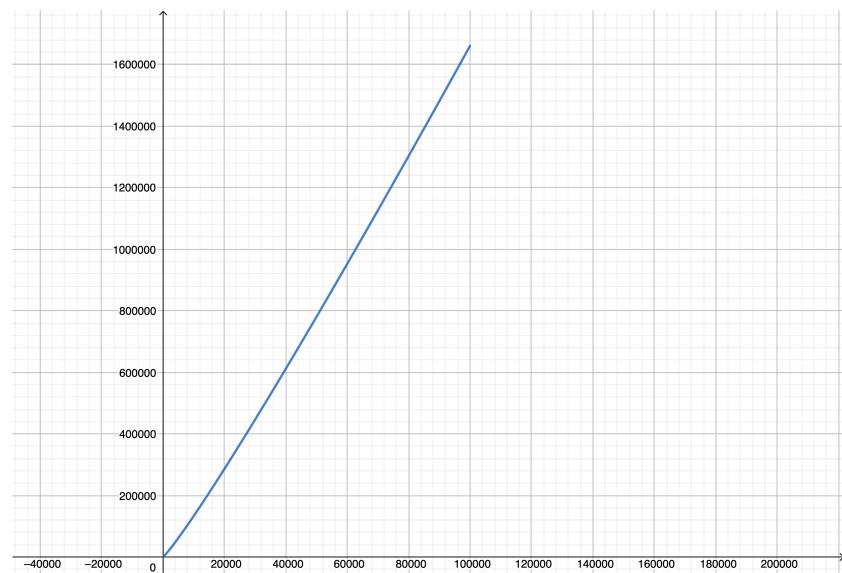
- $O(\log n)$ Complejidad logarítmica



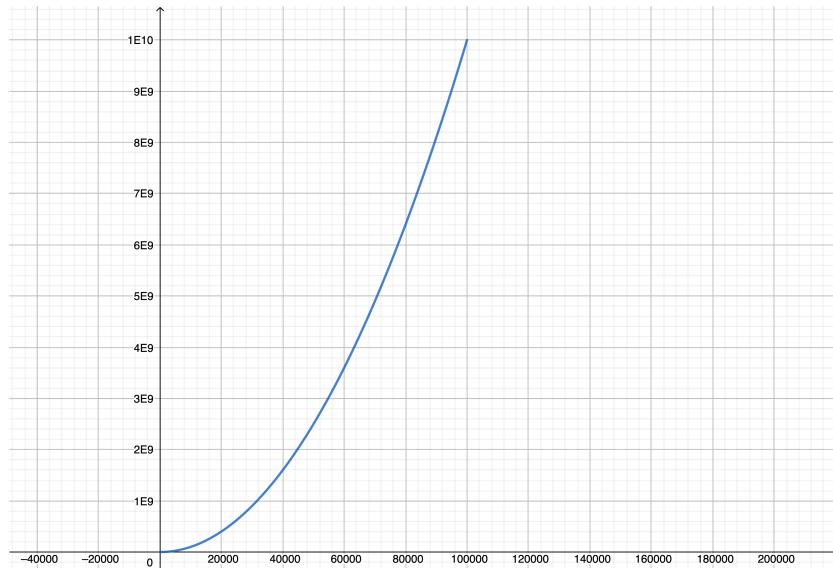
- $O(n)$ Complejidad lineal



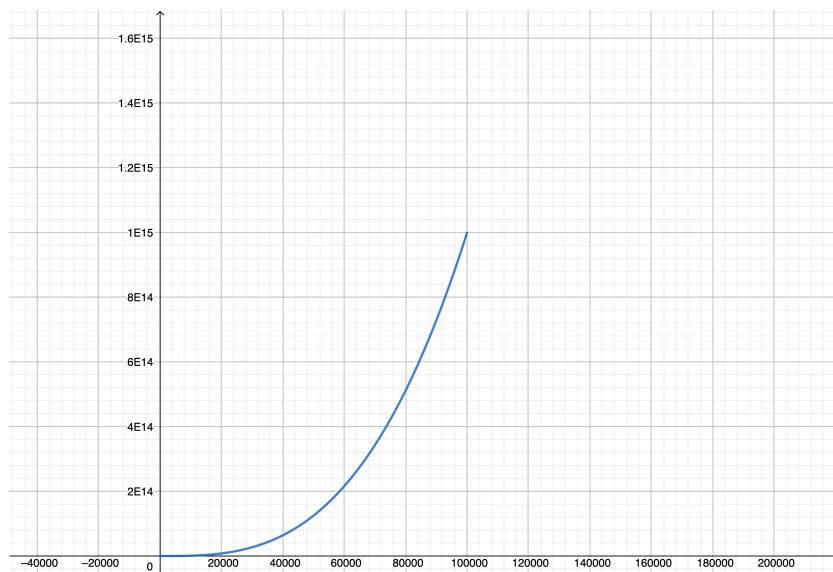
- $O(n \log n)$ Complejidad “n log n”



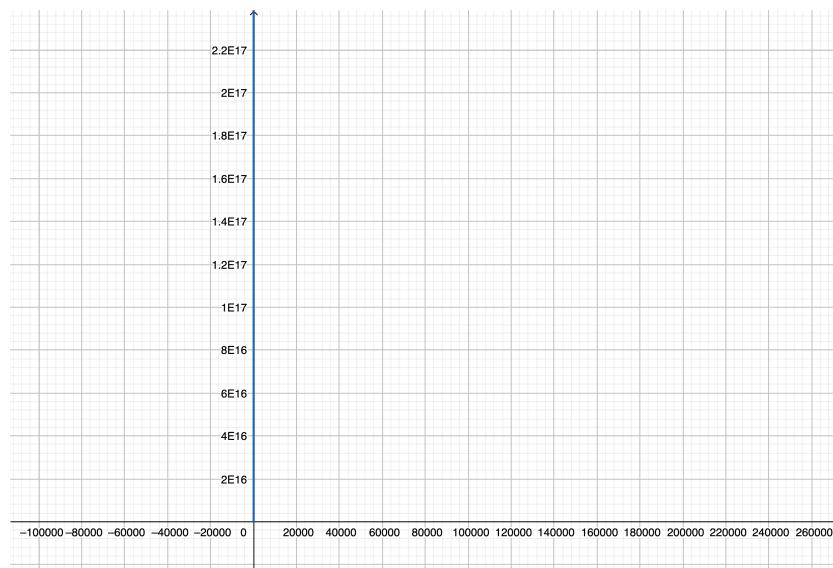
- $O(n^2)$ Complejidad cuadrática



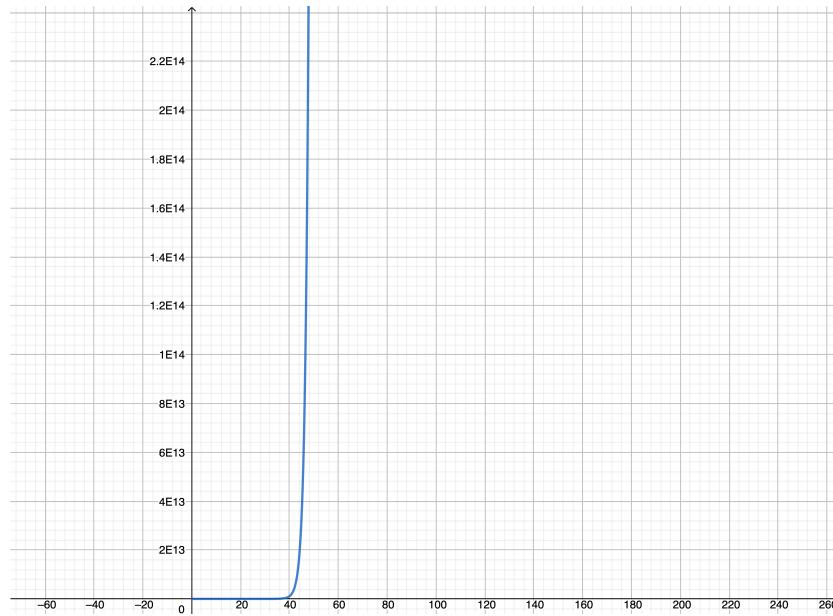
- $O(n^3)$ Complejidad cubica



- $O(c^n)$; $c > 1$ Complejidad exponencial

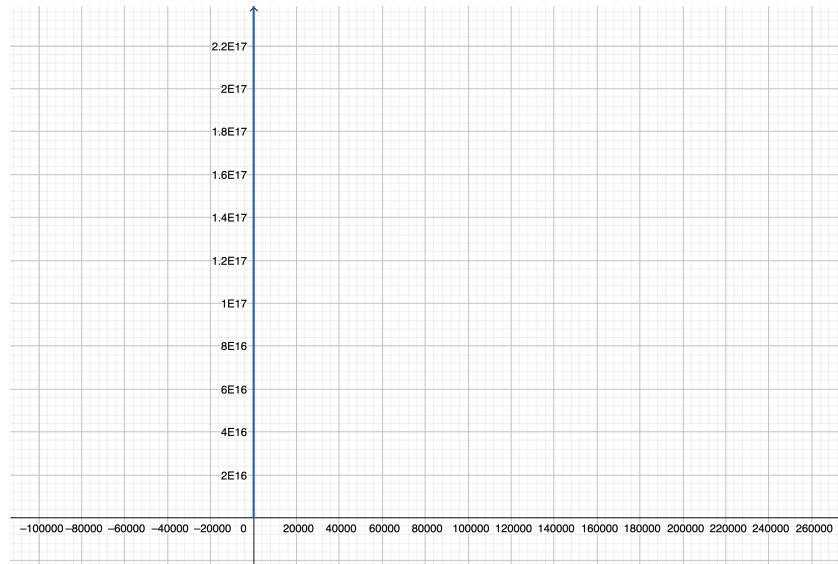


Los resultados arrojados cuando nuestra c toma el valor de 2, son increíblemente altos, pues aún haciendo una escala grande para "y" no fue posible para la graficadora geogebra arrojar la gráfica resultante, por lo cual nos acercamos a nuestros puntos más pequeños en "x", y esto fue lo se encontró:

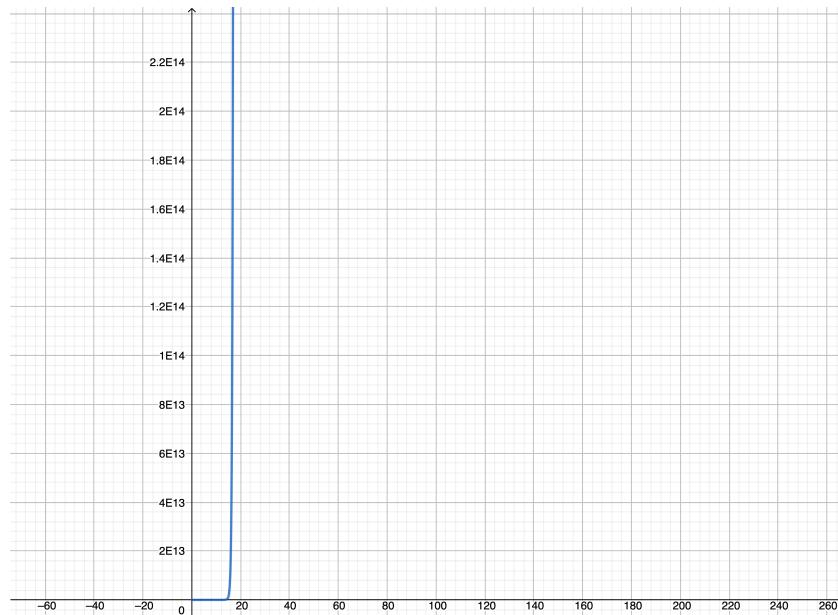


A partir de valores superiores a 40, nuestros valores se disparan demasiado, por lo que es inimaginable los valores que puede alcanzar a tener cuando nuestra x tiende a 100,000.

- $O(n!)$ Complejidad factorial



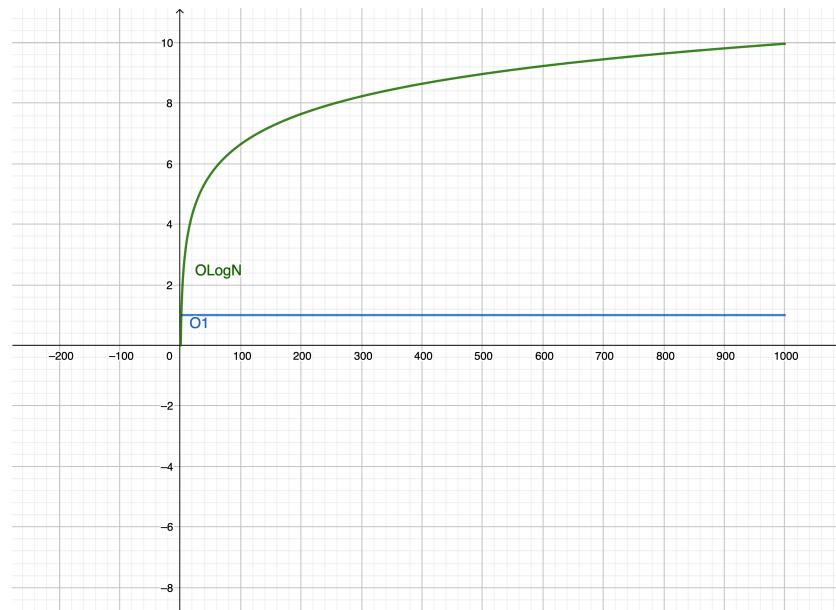
Al igual que en la complejidad anterior, la graficadora geogebra nos impide observar nuestra gráfica resultante debido a los valores tan extraordinariamente gigantescos que se obtienen aún cuando “ x ” toma valores chicos. Debido a esto, nos acercamos a los puntos de “ x ” en los que aún se puede distinguir cuánto vale “ y ”:



En cuanto nos comenzamos a acercar al valor de “ x ” en 20, nuestros valores de “ y ” se disparan demasiado, es debido a esto que es imposible para geogebra darnos valores de “ x ” mayores.

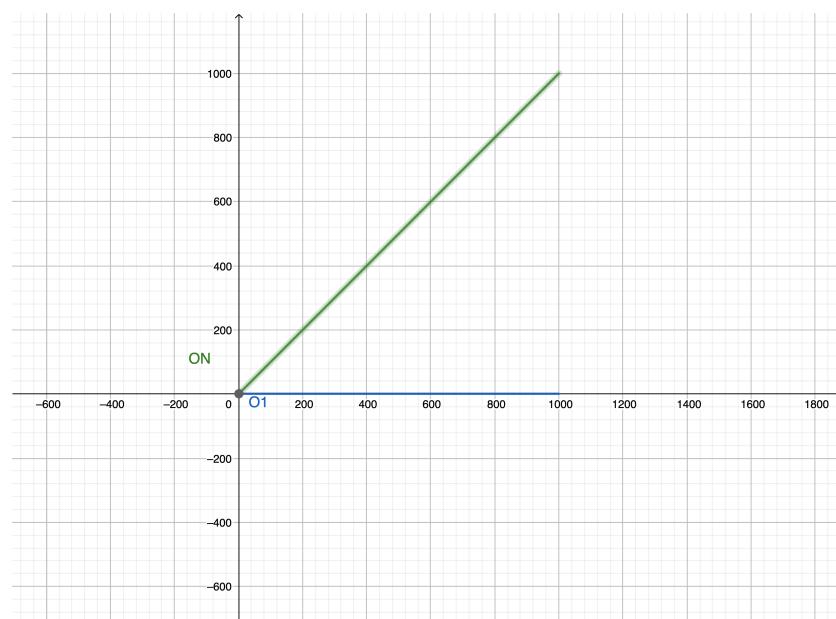
2. Confronte en pares a todos los ordenes en un rango de $0 < n < 1,000$ y de una justificación de cuál elegiría según cada par confrontado.

- $O(1)$ Complejidad constante vs $O(\log n)$ Complejidad logarítmica



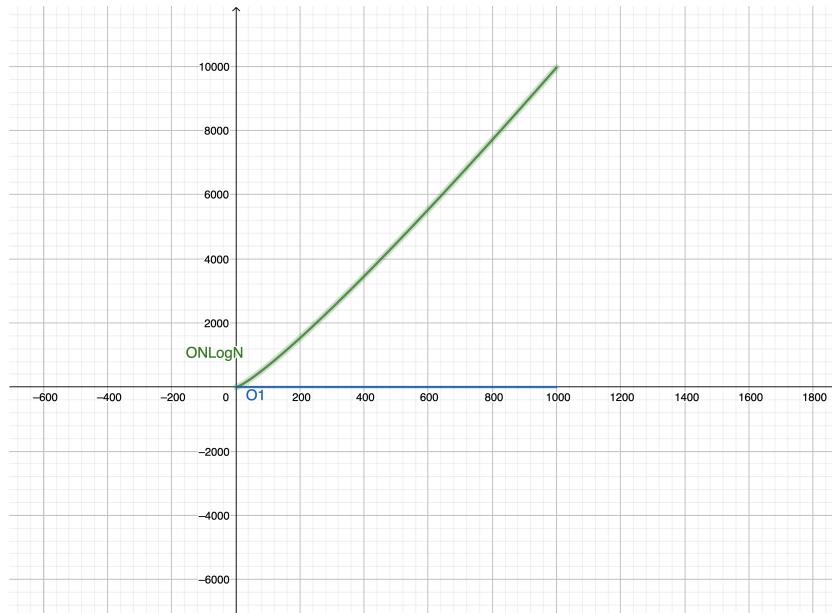
Se elige la complejidad constante $O(1)$ debido a que, aunque cuando el tamaño de problema es menor a 1 es más rápido $O(\log N)$, cuando crece el problema, $O(1)$ siempre se mantiene con el valor de 1 y $O(\log N)$ cada vez crece más en comparación a $O(1)$.

- $O(1)$ Complejidad constante vs $O(n)$ Complejidad lineal



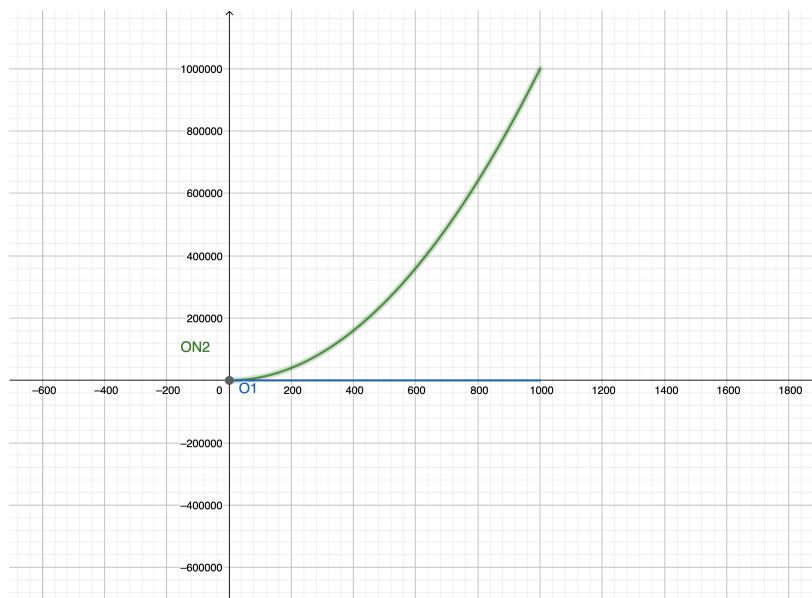
Se elige la complejidad constante $O(1)$, pues se puede apreciar que conforme la N va creciendo, $O(N)$ domina a $O(1)$ cada vez más

- $O(1)$ Complejidad constante vs $O(n \log n)$ Complejidad “n log n”



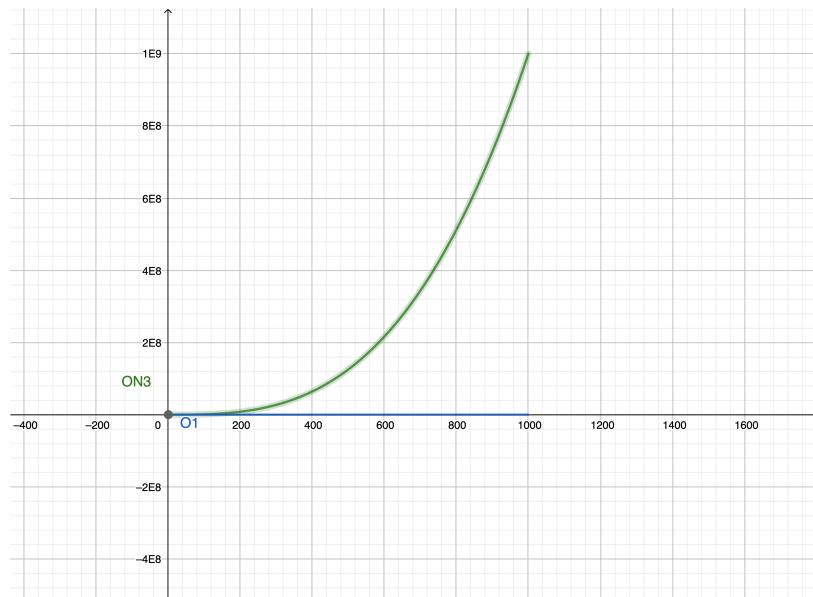
Se elige la complejidad constante $O(1)$, pues se puede apreciar que conforme la N va creciendo, $O(N \log N)$ domina a $O(1)$ cada vez más

- $O(1)$ Complejidad constante vs $O(n^2)$ Complejidad cuadrática



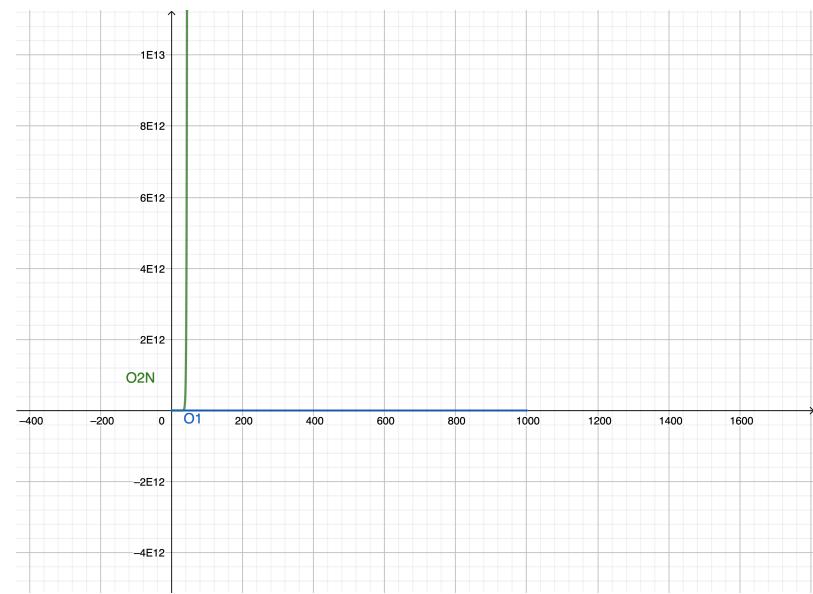
Se elige la complejidad constante $O(1)$, pues se puede apreciar que conforme la N va creciendo, $O(N^2)$ domina a $O(1)$ cada vez más

- $O(1)$ Complejidad constante vs $O(n^3)$ Complejidad cubica



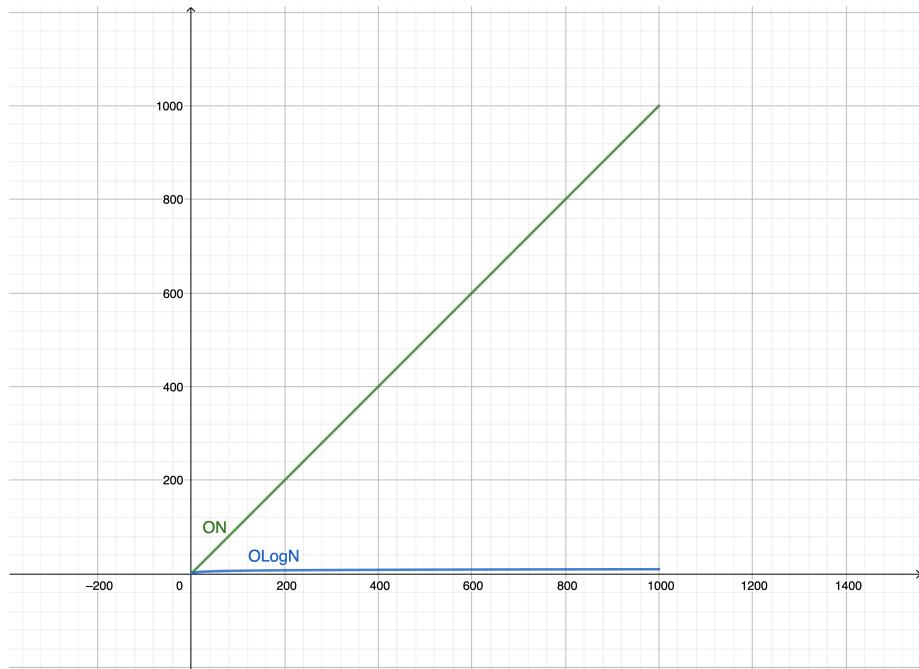
Se elige la complejidad constante $O(1)$, pues se puede apreciar que conforme la N va creciendo, $O(N^3)$ domina a $O(1)$ cada vez más

- $O(1)$ Complejidad constante vs $O(c^n)$; $c > 1$ Complejidad exponencial



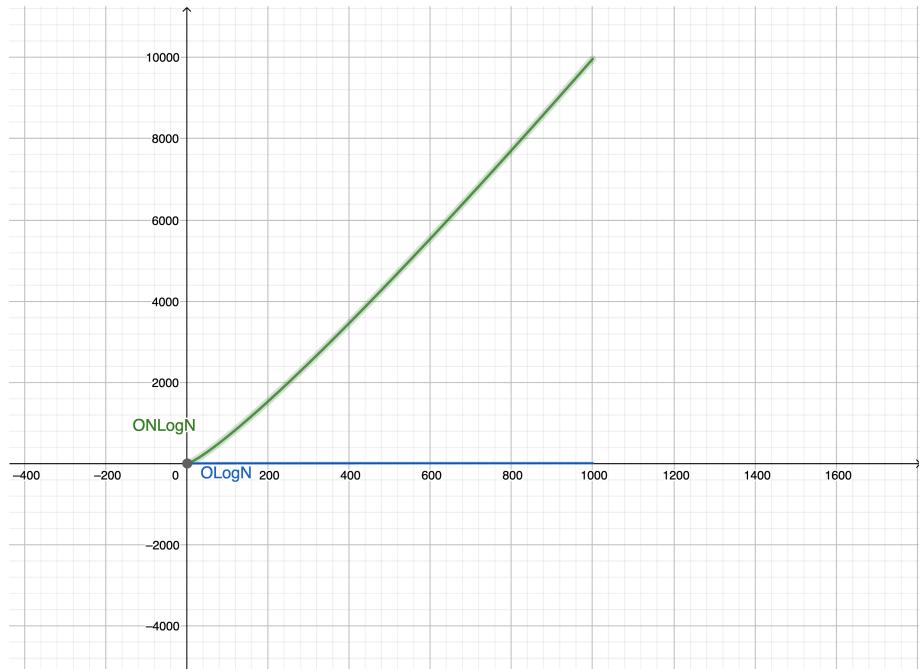
Se elige la complejidad constante $O(1)$, pues se puede apreciar que conforme la N va creciendo, $O(2^N)$ domina a $O(1)$ cada vez más

- $O(\log n)$ Complejidad logarítmica vs $O(n)$ Complejidad lineal



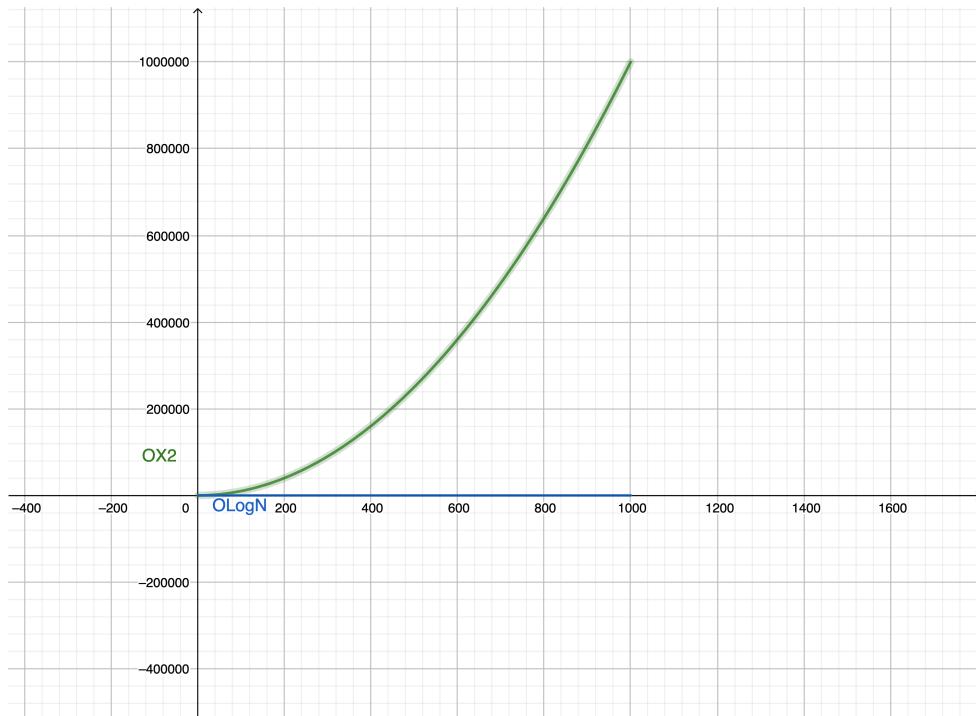
Se elige la complejidad logarítmica $O(\log N)$ pues en todo momento el tiempo de éste es menor que el de $O(N)$, llegando al punto en el que cuando el tamaño de problema es 1000, hay una diferencia de casi 1000 de tiempo.

- $O(\log n)$ Complejidad logarítmica vs $O(n \log n)$ Complejidad “n log n”



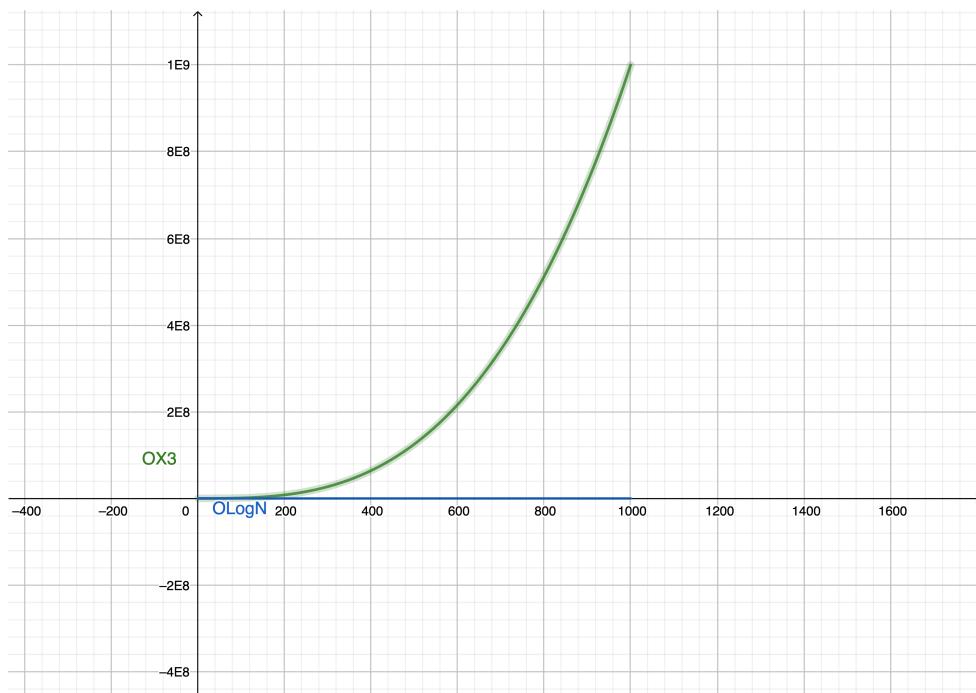
Se elige la complejidad constante $O(\log N)$, pues se puede apreciar que conforme la N va creciendo, $O(N \log N)$ domina a $O(\log N)$ cada vez más

- $O(\log n)$ Complejidad logarítmica vs $O(n^2)$ Complejidad cuadrática



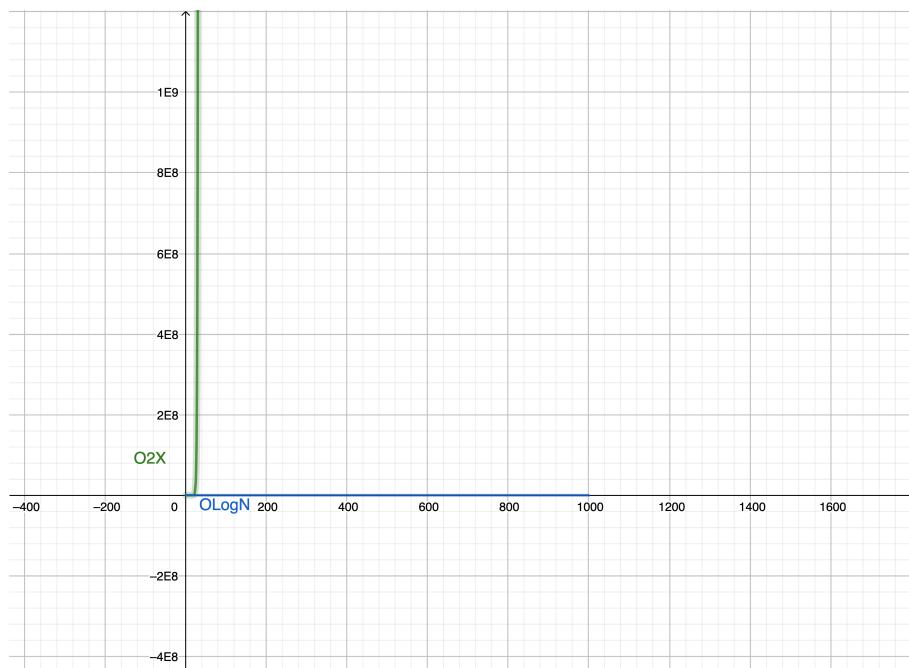
Se elige la complejidad constante $O(\log N)$, pues se puede apreciar que conforme la N va creciendo, $O(N^2)$ domina a $O(\log N)$ cada vez más

- $O(\log n)$ Complejidad logarítmica vs $O(n^3)$ Complejidad cubica



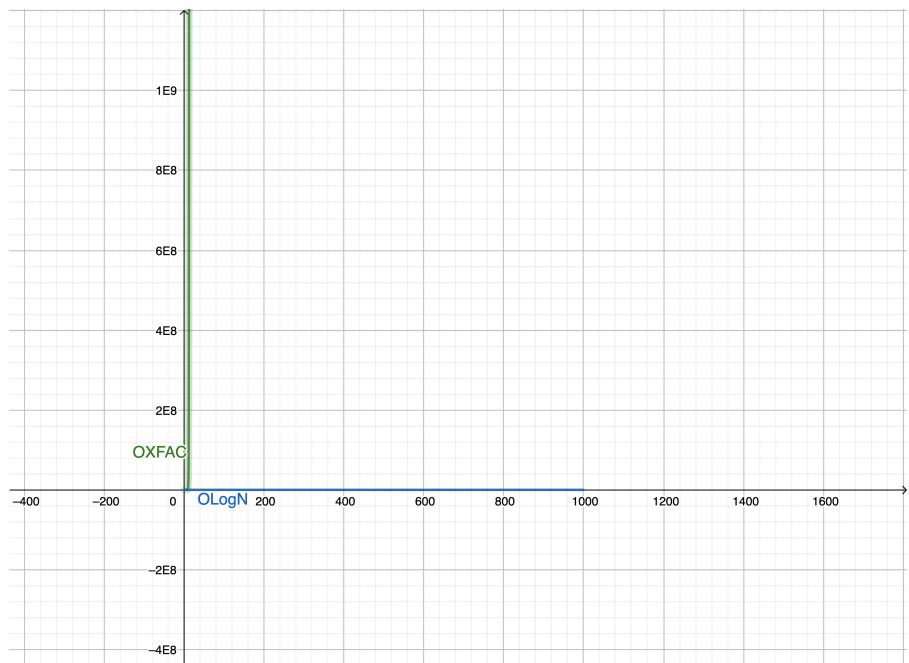
Se elige la complejidad constante $O(\log N)$, pues se puede apreciar que conforme la N va creciendo, $O(N^3)$ domina a $O(\log N)$ cada vez más

- $O(\log n)$ Complejidad logarítmica vs $O(c^n)$; $c > 1$ Complejidad exponencial



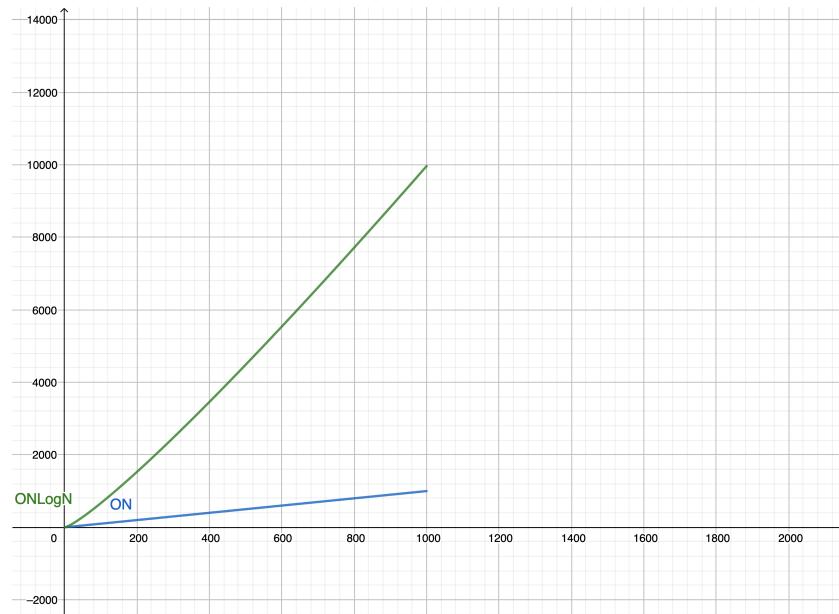
Se elige la complejidad constante $O(\log N)$, pues se puede apreciar que conforme la N va creciendo, $O(2^N)$ domina a $O(\log N)$ cada vez más

- $O(\log n)$ Complejidad logarítmica vs $O(n!)$ Complejidad factorial



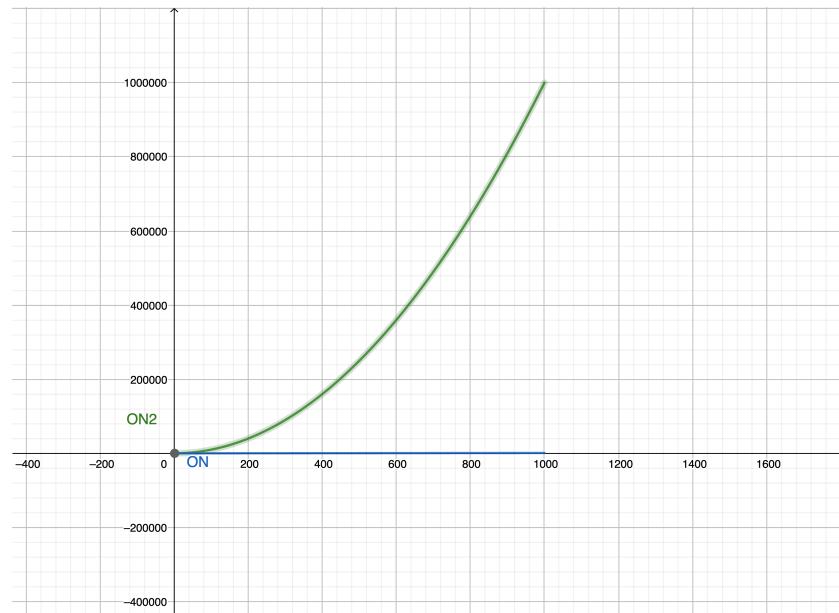
Se elige la complejidad constante $O(\log N)$, pues se puede apreciar que conforme la N va creciendo, $O(\log N)$ domina a $O(N!)$ cada vez más

- $O(n)$ Complejidad lineal vs $O(n \log n)$ Complejidad “n log n”



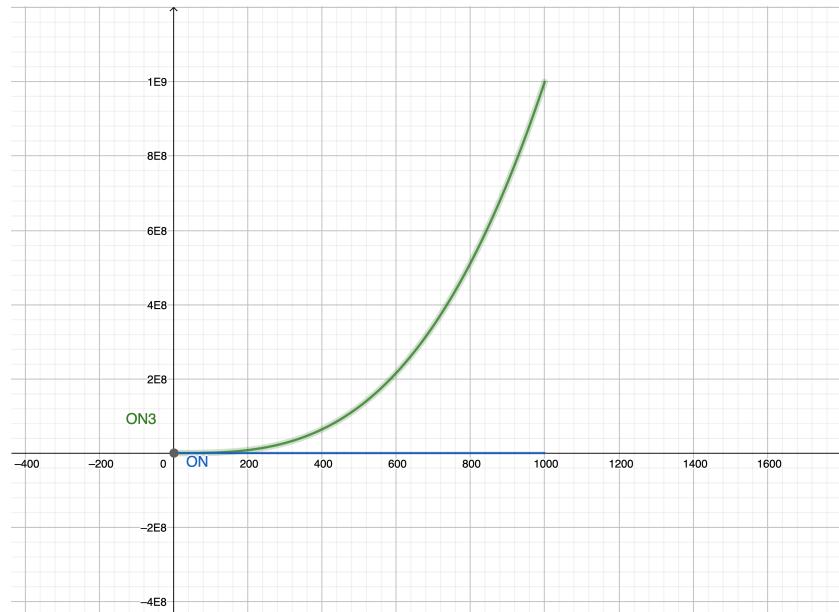
Se elige la complejidad lineal $O(N)$, ya que $O(N \log N)$ tiende a crecer mucho más conforme avanza el tamaño del problema, tanto así que cuando $N=1000$ la diferencia es aproximadamente 9,000 en tiempo.

- $O(n)$ Complejidad lineal vs vs $O(n^2)$ Complejidad cuadrática



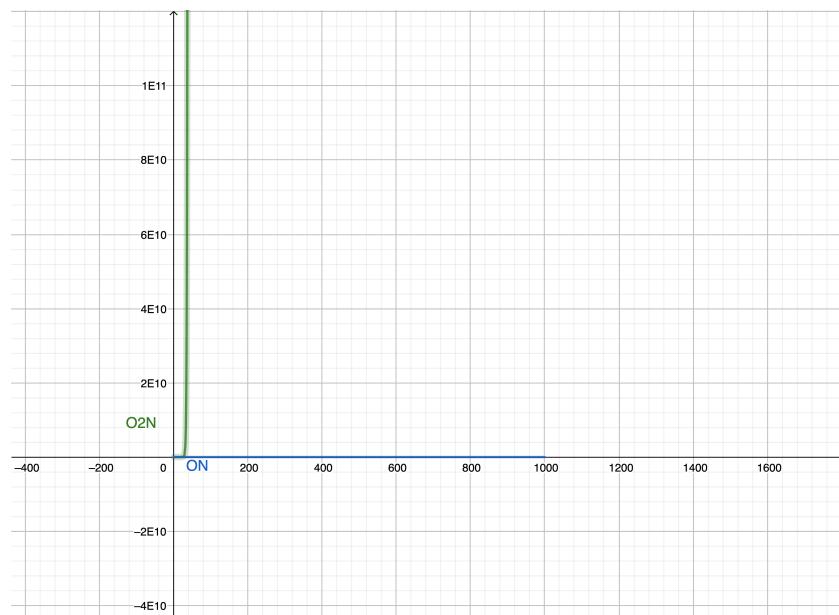
Se elige la complejidad constante $O(N)$, pues se puede apreciar que conforme la N va creciendo, $O(N^2)$ domina a $O(N)$ cada vez más

- $O(n)$ Complejidad lineal vs vs $O(n^3)$ Complejidad cubica



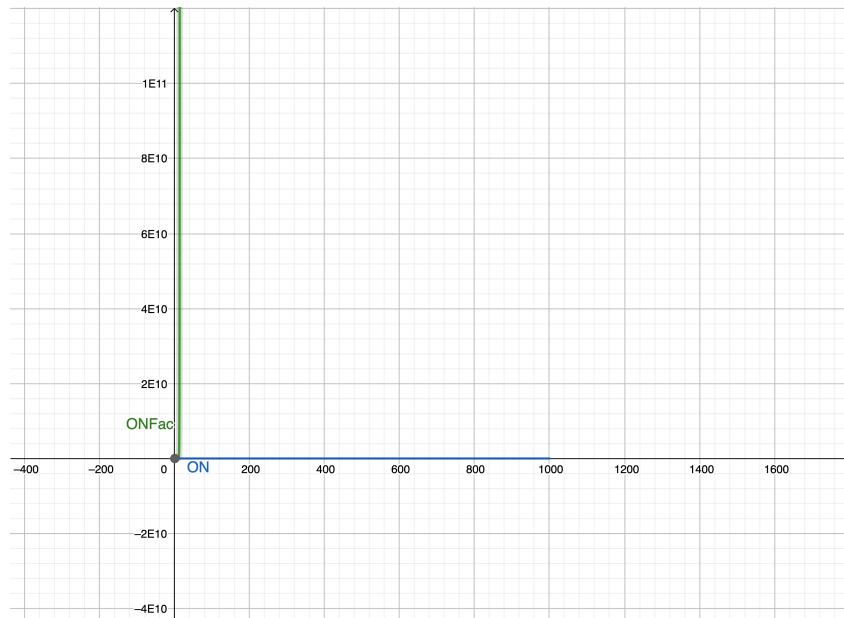
Se elige la complejidad constante $O(N)$, pues se puede apreciar que conforme la N va creciendo, $O(N^3)$ domina a $O(N)$ cada vez más

- $O(n)$ Complejidad lineal vs vs $O(c^n)$; $c > 1$ Complejidad exponencial



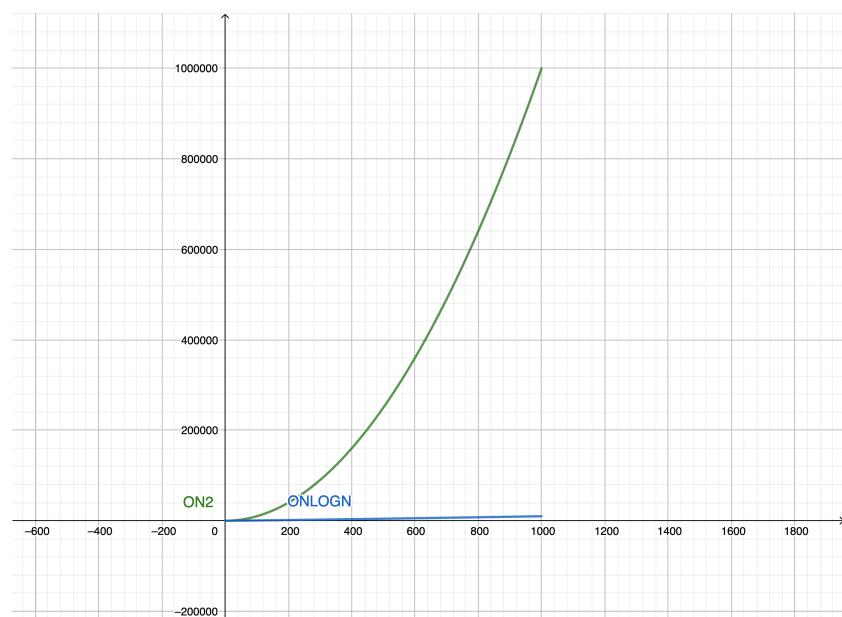
Se elige la complejidad constante $O(N)$, pues se puede apreciar que conforme la N va creciendo, $O(2^N)$ domina a $O(N)$ cada vez más

- $O(n)$ Complejidad lineal vs $O(n!)$ Complejidad factorial



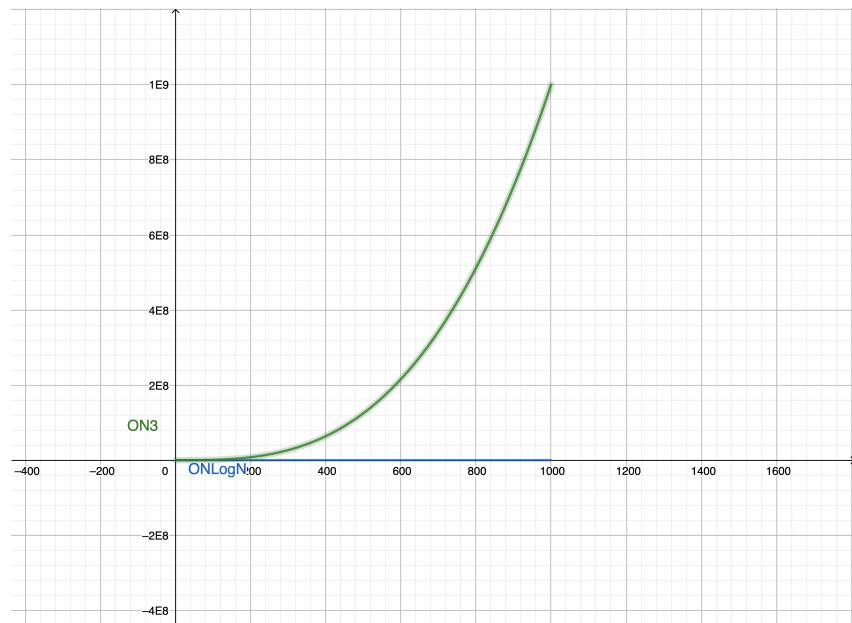
Se elige la complejidad constante $O(N)$, pues se puede apreciar que conforme la N va creciendo, $O(N!)$ domina a $O(N)$ cada vez más

- $O(n \log n)$ Complejidad “n log n” vs $O(n^2)$ Complejidad cuadrática



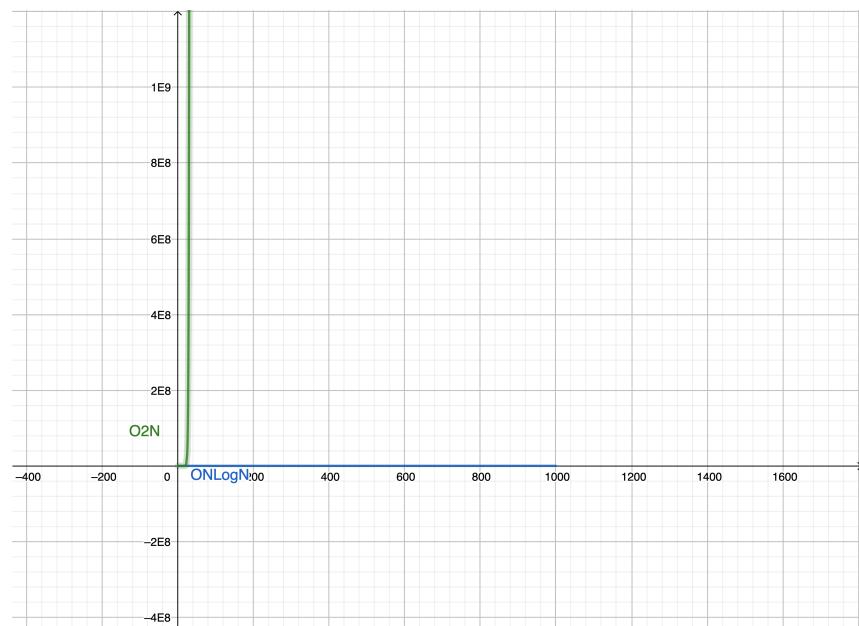
Se elige la complejidad $O(N \log N)$, pues el tiempo que tarda en cada uno de los tamaños de problema son mucho más pequeños en comparación con $O(N^2)$, tanto así que cuando el tamaño de problema está alrededor de 150 ya tarda más que $O(N \log N)$ evaluado en 1000

- $O(n \log n)$ Complejidad “ $n \log n$ ” vs $O(n^3)$ Complejidad cubica



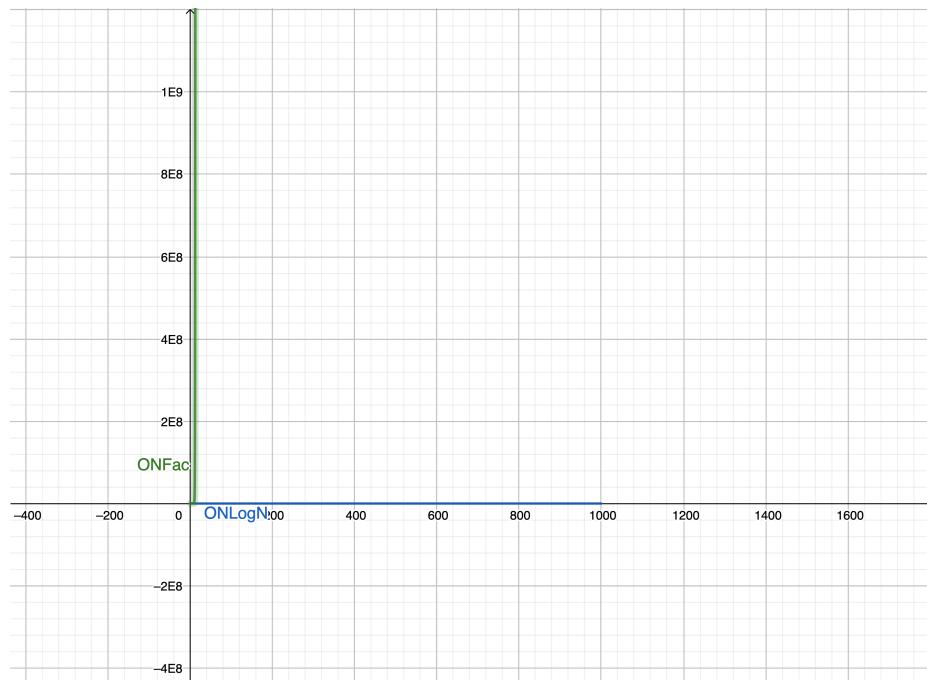
Se elige la complejidad constante $O(N \log N)$, pues se puede apreciar que conforme la N va creciendo, $O(N^3)$ domina a $O(N \log N)$ cada vez más

- $O(n \log n)$ Complejidad “ $n \log n$ ” vs $O(c^n)$; $c > 1$ Complejidad exponencial



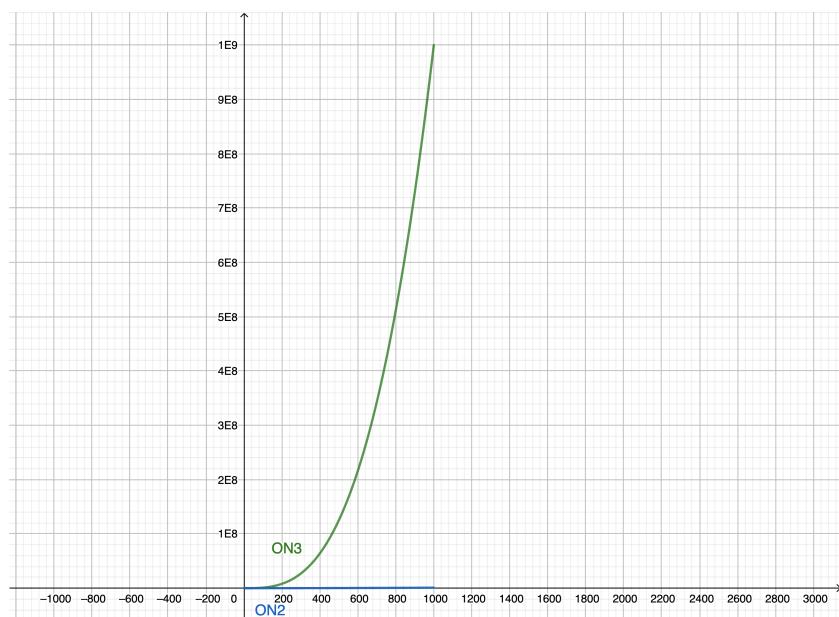
Se elige la complejidad constante $O(N \log N)$, pues se puede apreciar que conforme la N va creciendo, $O(2^N)$ domina a $O(N \log N)$ cada vez más

- $O(n \log n)$ Complejidad “ $n \log n$ ” vs $O(n!)$ Complejidad factorial



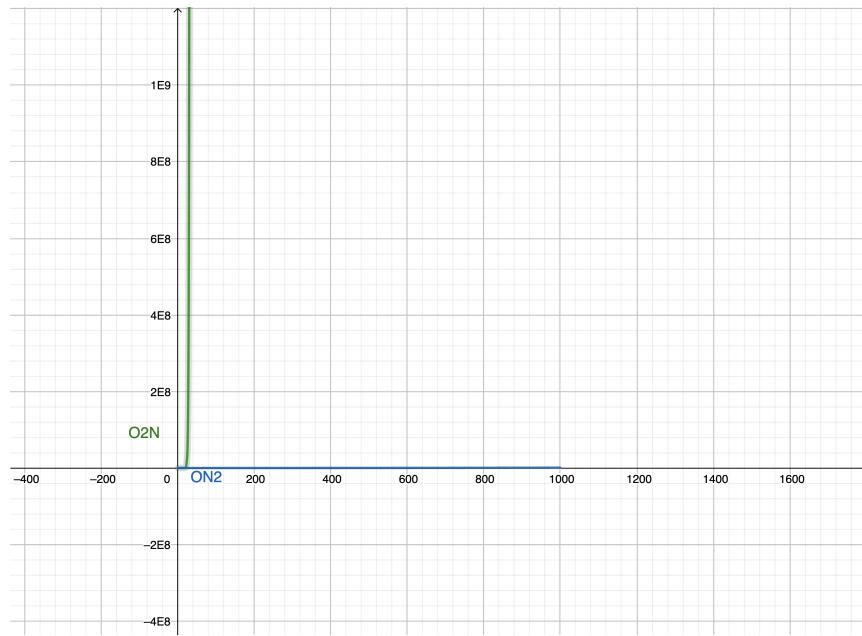
Se elige la complejidad constante $O(N \log N)$, pues se puede apreciar que conforme la N va creciendo, $O(N!)$ domina a $O(N \log N)$ cada vez más

- $O(n^2)$ Complejidad cuadrática vs $O(n^3)$ Complejidad cúbica



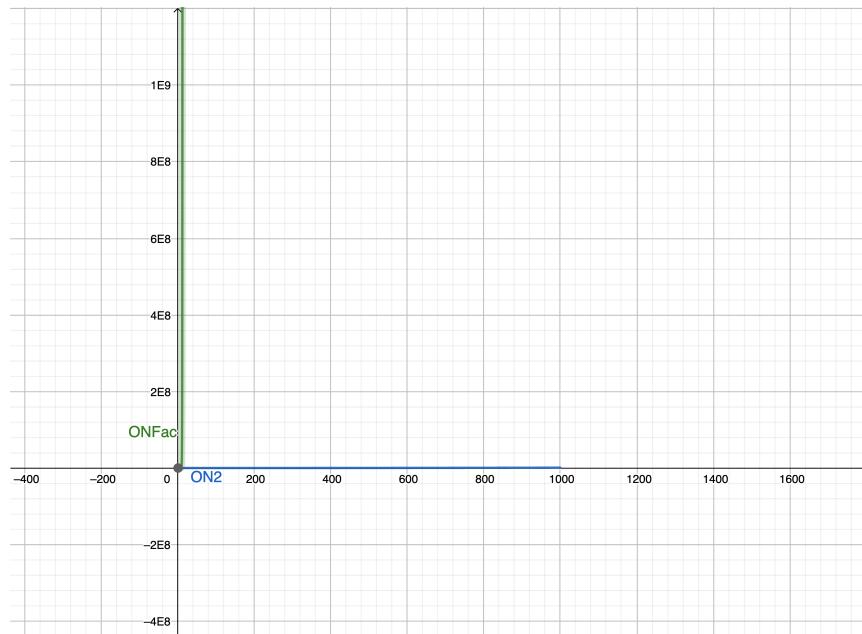
Se elige la complejidad lineal $O(N^2)$, pues la gráfica nos muestra que mientras más grande sea el valor de N, la complejidad $O(N^3)$ se despega considerablemente de $O(N^2)$.

- $O(n^2)$ Complejidad cuadrática vs $O(c^n)$; $c > 1$ Complejidad exponencial



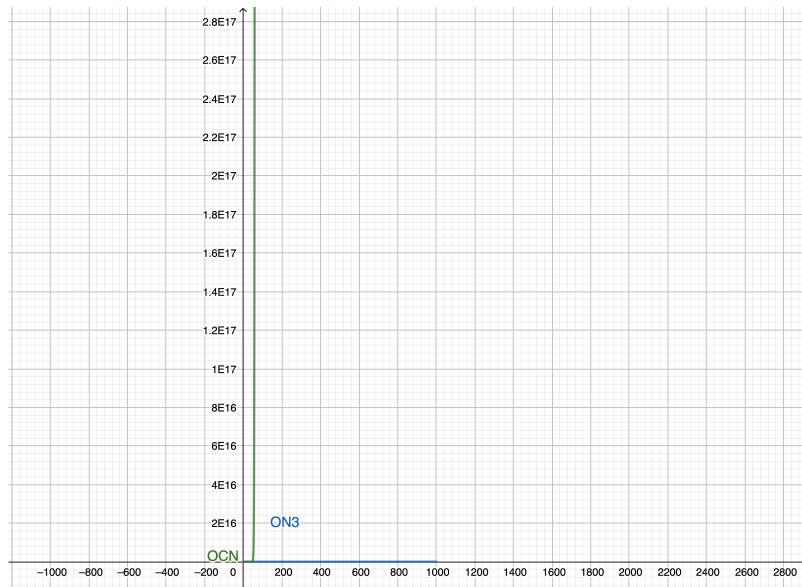
Se elige la complejidad constante $O(N^2)$, pues se puede apreciar que conforme la N va creciendo, $O(2^N)$ domina a $O(N^2)$ cada vez más

- $O(n^2)$ Complejidad cuadrática vs $O(n!)$ Complejidad factorial

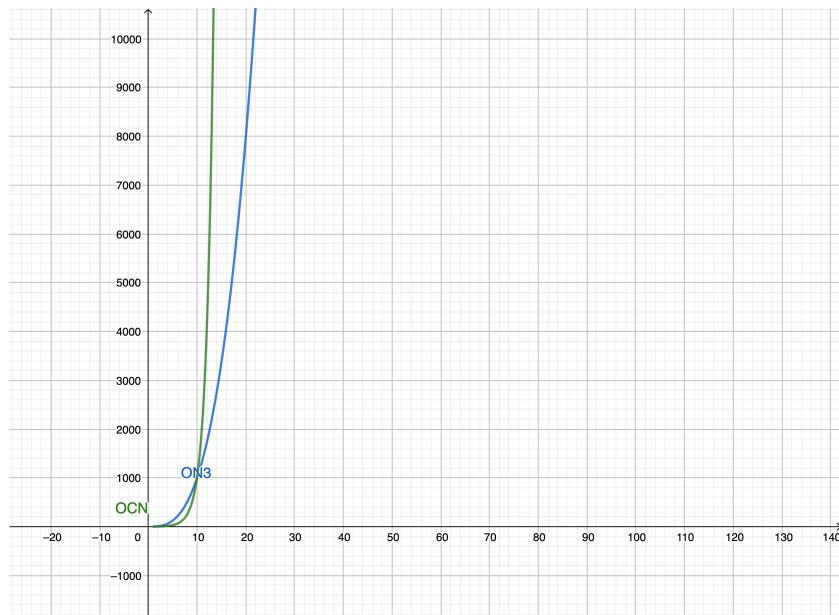


Se elige la complejidad constante $O(N^2)$, pues se puede apreciar que conforme la N va creciendo, $O(N!)$ domina a $O(N^2)$ cada vez más

- $O(n^3)$ Complejidad cúbica vs $O(c^n)$; $c > 1$ Complejidad exponencial

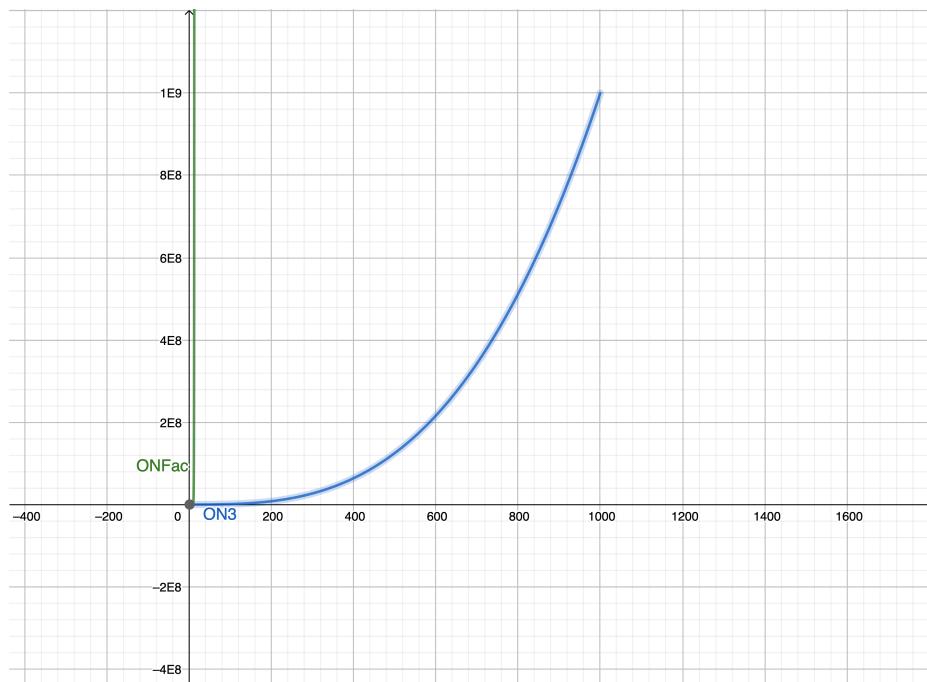


Es increíble la diferencia tan abismal que existe entre las dos complejidades, pues aunque en la complejidad exponencial el valor de C que se maneja es igual a 2, antes de llegar a una décima parte del tamaño de problema 1,000 es imposible para la graficadora mostrarnos el tiempo de dicha complejidad, es por esto que se elige la complejidad $O(N^3)$.



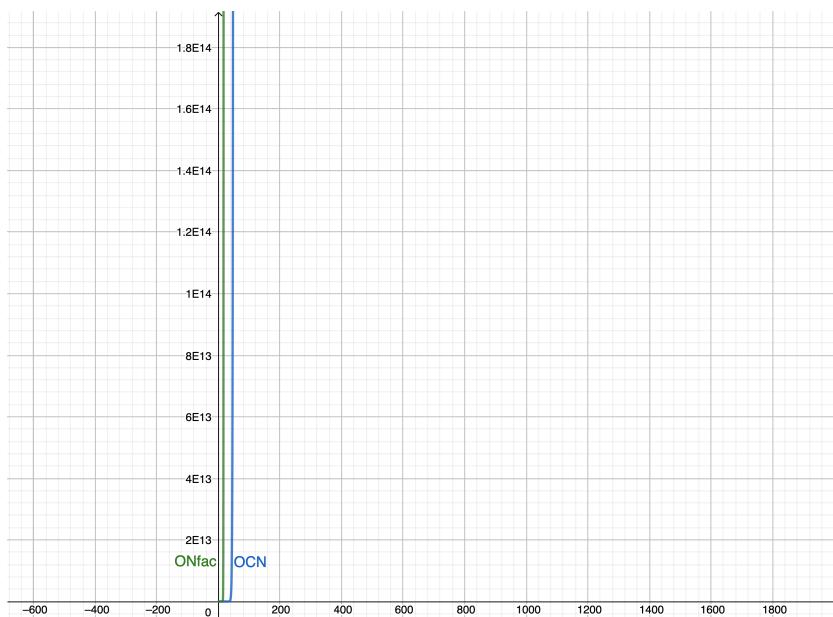
Se observan los valores que tienen las dos complejidades de con los tamaños de problema de 0 a 20, y se puede notar que aunque $O(2^N)$ es menor a $O(N^3)$, aproximadamente después del tamaño 10, comienzan a darse estas diferencias tan grandes en los tiempos, es por esto que elegimos $O(N^3)$, además que mientras la C tome valores mayores, $O(N^3)$ comenzará ser mejor opción desde antes del tamaño de problema 10.

- $O(n^3)$ Complejidad cúbica vs $O(n!)$ Complejidad factorial

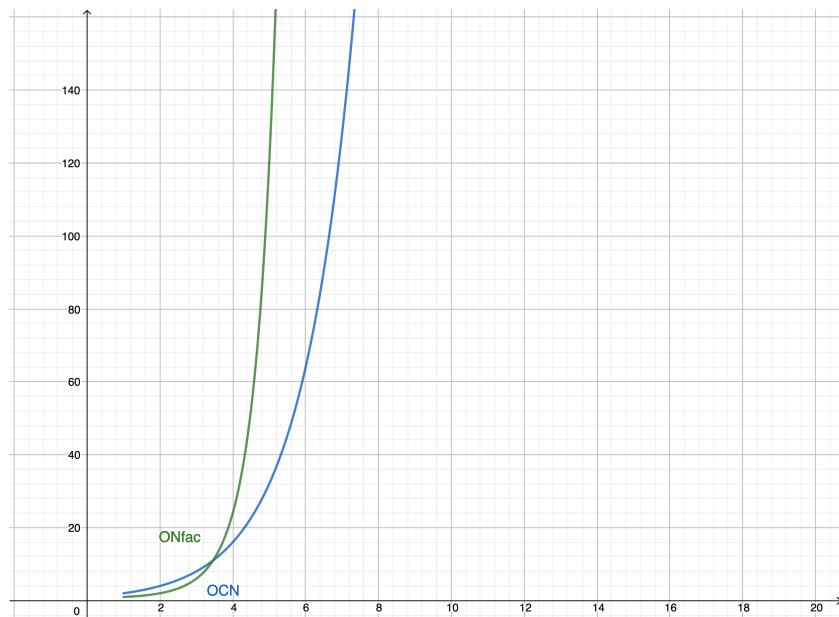


Se elige la complejidad constante $O(N^3)$, pues se puede apreciar que conforme la N va creciendo, $O(N!)$ domina a $O(N^3)$ cada vez más

- $O(c^n)$; $c > 1$ Complejidad exponencial vs $O(n!)$ Complejidad factorial



Las dos complejidades son terriblemente grandes, pues el tiempo que tardan conforme avanza el tamaño de problema crece a pasos agigantados. Sin embargo se puede notar en la gráfica que $O(2^n)$ es más rápida que $O(N!)$, pues este último se dispara con un tamaño de problema más chico que cuando se dispara $O(2^n)$.



Nos acercamos y notamos que aproximadamente cuando N vale 3, $O(2^N)$ se vuelve una mejor opción y $O(N!)$ comienza a tener valores muy grandes, esto puede cambiar si le damos valores más grandes a C, sin embargo conforme aumente N se distinguirá que $O(C^N)$ es mejor opción a elegir.