Matemáticas Discretas

Oscar Bedoya

oscar.bedoya@correounivalle.edu.co

http://eisc.univalle.edu.co/~oscarbed/MD/

* Notación O

Crecimiento de funciones

Donald Knuth

• Cuando estaba en 8° grado participó en un concurso que consistía en formas palabras con las letras de la expresión "Ziegler's giant Bar"

• Estudió Física, matemáticas y ciencias

• Escribió *The Art of Computer Programming*

•Desarrolló TeX Latex Documentos

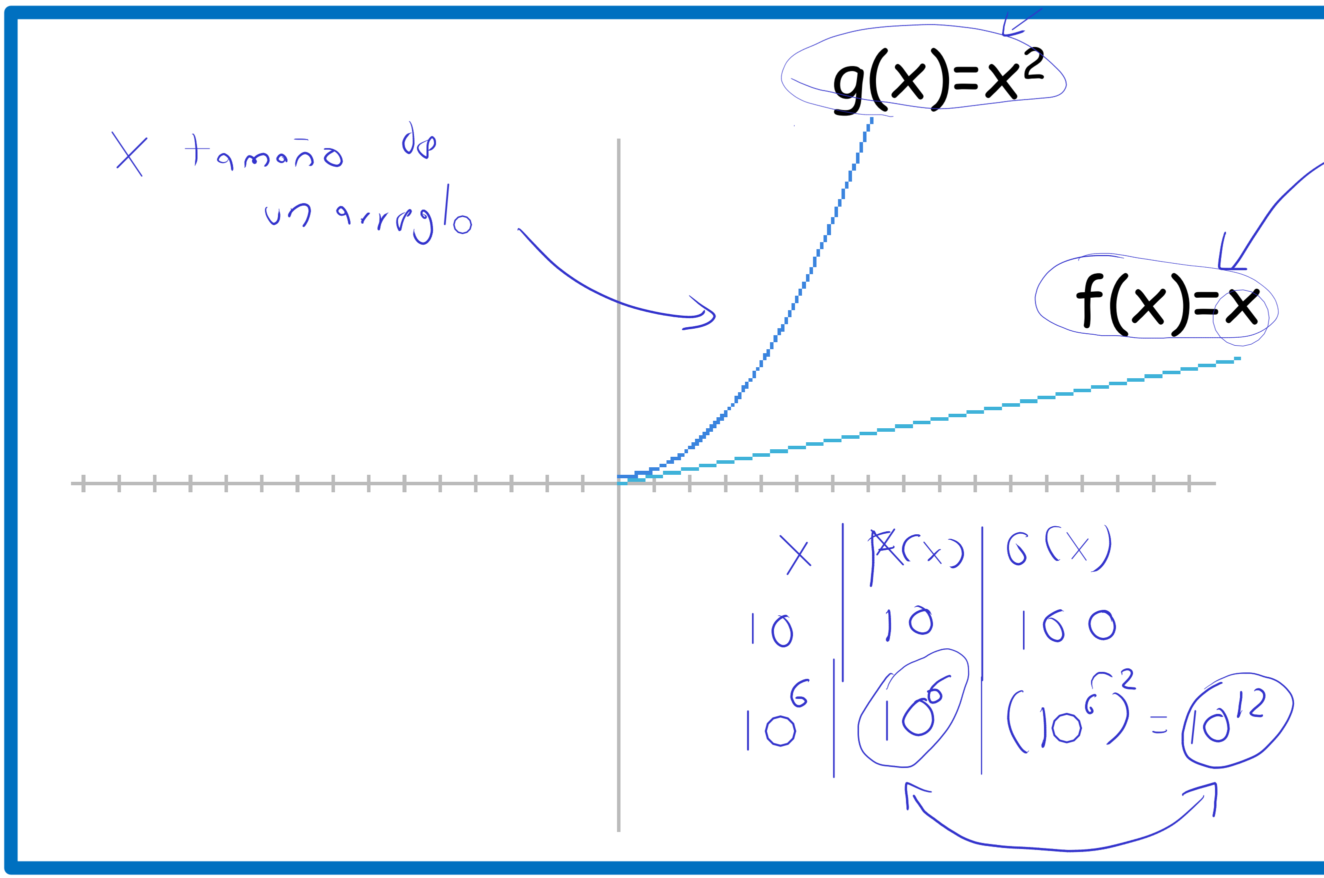
(1938 -)

Crecimiento de funciones

El **análisis de crecimiento de funciones** se basa en comparar el comportamiento de dos o más funciones

$\boxed{g(x)}$ ALG 1

$\boxed{f(x)}$ ALG 2



X	f(x)	G(x)
10	10	100
10^6	10^6	$(10^6)^2 = 10^{12}$

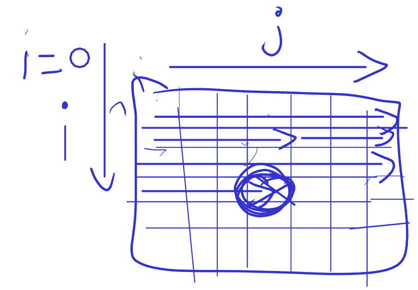
Complejidad Computacional

1'000.000 veces

Considere el siguiente programa que busca el número **b** en una matriz:

Funcion(int b)

datos[i][j]




```
for (int i=0; i<10 ; i=i+1){
  for (int j=0; j<10 ; j=j+1){
    if (datos[i][j]==b){
      System.out.println("Encontrado");
      break;
    }
  }
}
```

Considere el siguiente programa que busca el número **b** en una matriz:

1	2	3	4	5	6	7	8	9	10
-8	12	-9	65	34	56	-54	55	14	7
-9	34	90	56	87	94	31	45	88	99
43	98	34	65	54	65	52	54	76	55
14	13	31	46	15	9	41	-8	11	13
89	32	23	13	41	23	77	80	88	-9
19	56	34	23	25	55	99	14	11	56
14	34	56	43	-78	25	77	25	22	95
94	-6	94	14	80	-6	67	95	-8	-76
89	80	-34	76	89	-4	13	11	78	15
-56	-56	13	95	-87	77	99	87	77	-3

10x10

b=7

Considere el siguiente programa que busca el número **b** en una matriz:

1

7	12	-9	65	34	56	-54	55	14	-65
-34	34	90	56	87	94	31	45	88	99
43	98	34	65	54	65	52	54	76	55
14	13	31	46	15	9	41	-8	11	13
89	32	23	13	41	23	77	80	88	-9
19	56	34	23	25	55	99	14	11	56
14	34	56	43	-78	25	77	25	22	95
94	-6	94	14	80	-6	67	95	-8	-76
89	80	-34	76	89	-4	13	11	78	15
-56	-56	13	95	-87	77	99	87	77	-3

10x10

b=7

Considere el siguiente programa que busca el número **b** en una matriz:

-8	12	-9	65	34	56	-54	55	14	-65
-35	34	90	56	87	94	31	45	88	99
43	98	34	65	54	65	52	54	76	55
14	13	31	46	15	9	41	-8	11	13
89	32	23	13	41	23	77	80	88	-9
19	56	34	23	25	55	99	14	11	56
14	34	56	43	-78	25	77	25	22	95
94	-6	94	14	80	-6	67	95	-8	-76
89	80	-34	76	89	-4	13	11	78	15
7	-56	13	95	-87	77	99	87	77	-3

91

10x10

b=7

Considere el siguiente programa que busca el número **b** en una matriz:

-8	12	-9	65	34	56	-54	55	14	-65
-35	34	90	56	87	94	31	45	88	99
43	98	34	65	54	65	52	54	76	55
14	13	31	46	15	9	41	-8	11	13
89	32	23	13	41	23	77	80	88	-9
19	56	34	23	25	55	99	14	11	56
14	34	56	43	-78	25	77	25	22	95
94	-6	94	14	80	-6	67	95	-8	-76
89	80	-34	76	89	-4	13	11	78	15
32	-56	13	95	-87	77	99	87	77	7

100

10x10

b=7

Considere el siguiente programa que busca el número **b** en una matriz:

```
for (int i=0; i<10 ; i=i+1){
   for (int j=0; j<10 ; j=j+1){
      if (datos[i][j]==b){
         System.out.println("Encontrado");
         break;
      }
   }
}
```

1) Tamaño de la matriz
2) La posición que se encuentra el número

¿De que depende el número de pasos que realiza el algoritmo?

mejor caso

menos pasos

caso promedio

analisis

peor caso

mas pasos

Considere el siguiente programa que busca el número **b** en una matriz:

```
for (int i=0; i<n ; i=i+1){
  for (int j=0; j<n ; j=j+1){
    if (datos[i][j]==b){
      System.out.println("Encontrado");
      break;
    }
  }
}
```

$\boxed{F(n) = n^2}$

n

1) Final

2) No existe

En el peor caso, ¿cuántas comparaciones hará el programa para una matriz de tamaño nxn?

Problema: Dado un arreglo ordenado encontrar el valor b

$b = 7$

-3	-2	0	3	7	11	14	22	26	34

Problema: Dado un arreglo ordenado encontrar el valor b

-3	-2	0	3	7	11	14	22	26	34

b=7

```
for(int i=0; i<10; i++){
        if(datos[i]==b)
                print("Encontrado")
                break
}
```

mejor = 1

peor = 10

Problema: Dado un arreglo ordenado encontrar el valor b

-3	-2	0	3	7	11	14	22	26	34

b=34

Problema: Dado un arreglo ordenado encontrar el valor b

-3	-2	0	3	7	11	14	22	26	34

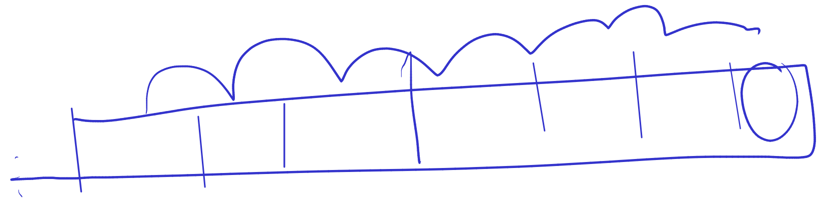
b=-3

Programa 1:

```
public void buscar(){
 for(int i=1; i<=n; i=i+1){
  if (datos[i]==b){
    System.out.println("Encontrado");
    break;
  }
 }
}
```

buscar = n

Programa 1:



```
public void buscar(){
  for(int i=1; i<=n; i=i+1){
   if (datos[i]==b){
     System.out.println("Encontrado");
     break;
   }
  }
}
```

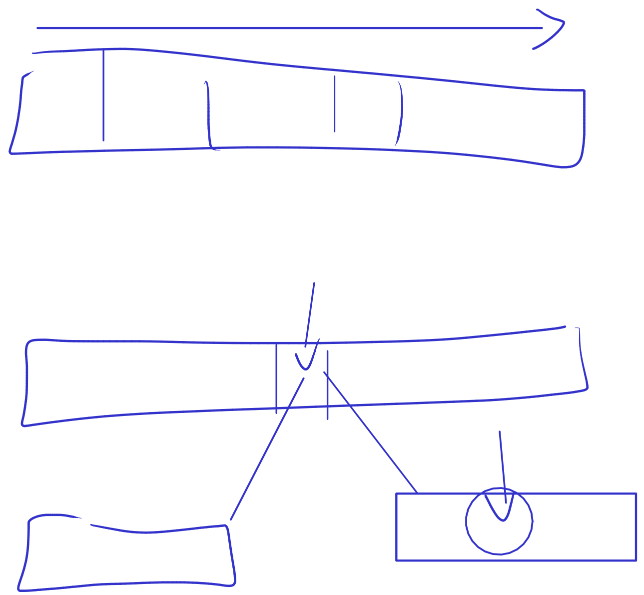
Tamaño n

peor caso = n

En el peor caso, ¿cuántas comparaciones hará el programa para un arreglo de tamaño n?

Programa 2:

```
public void buscar(int i, int j){
  medio=(i+j)/2;
  if (a[medio]==b){
    System.out.println("Encontrado");
    break;
  }
  if (a[medio]<b)
    buscar(medio,j);
  if (a[medio]>b)
    buscar(i,medio);
}
```



NO TENGO QUE HACER TODAS LAS COMPARACIONES CON EL ALGORITMO 2

IZQ DER

-3	-2	0	3	7	11	14	22	26	34

n

$b = 3$

Izq Der

14	22	26	34

$n/2$

$b = 34$

Divide y vencerás

ORDENADO

$n/2 = 5$

Encontrado

$\boxed{34}$

$n/4$

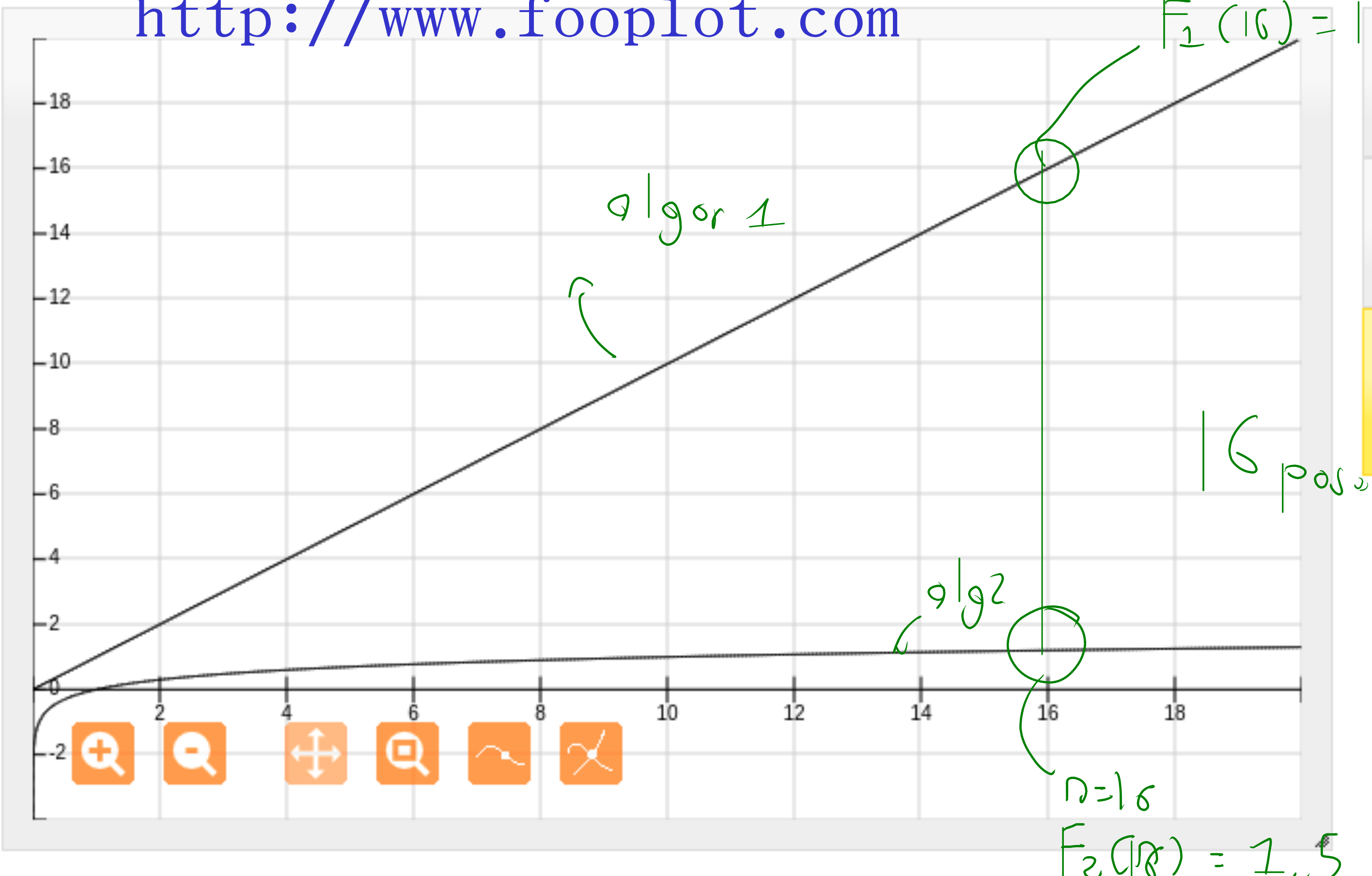
Programa 2:

```
public void buscar(int i, int j){
  medio=(i+j)/2;
  if (a[medio]==b){
    System.out.println("Encontrado");
    break;
  }
  if (a[medio]<b)
    buscar(medio,j);
  if (a[medio]>b)
    buscar(i,medio);
}
```

$\boxed{\log(n)}$

En el peor caso, ¿cuántas comparaciones hará el programa para un arreglo de tamaño n?

http://www.fooplot.com



Numero de compos por caso

16 pasos vs 1.5 pasos

log(n)

Complejidad computacional: Nümero pasos que requiere un algoritmo para solucionar un problema

Complejidad espacial: Memoria requerida*

no es un problema

n	$f1(n) = n$	$f2(n) = \log_2(n)$
	Fast	+ complicated
100	100	6.64
1000000	1000000	19.93
100000000	100000000	26.57

[illegible]

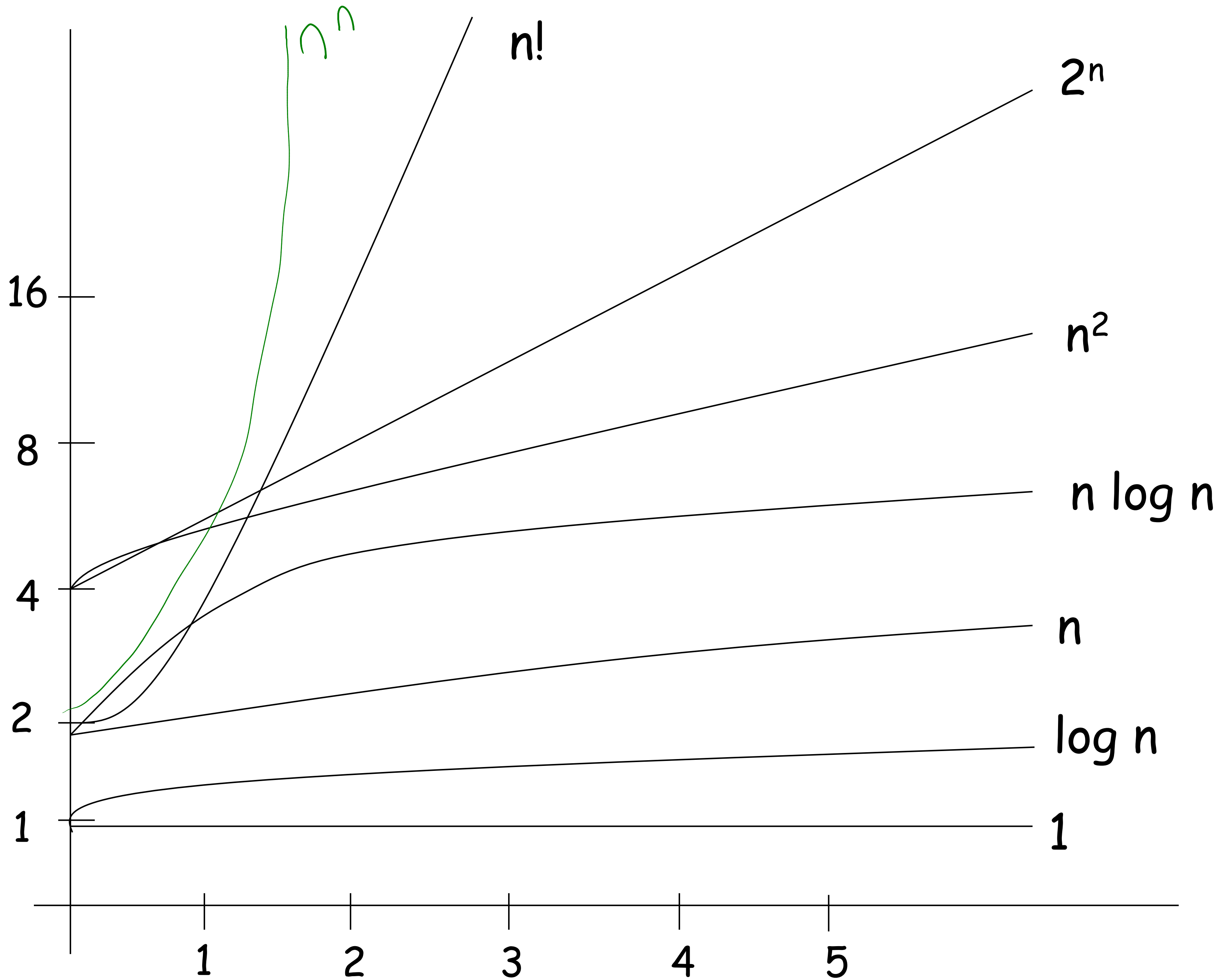
100000000 sec VS 20.87

26.57 s

27 sec

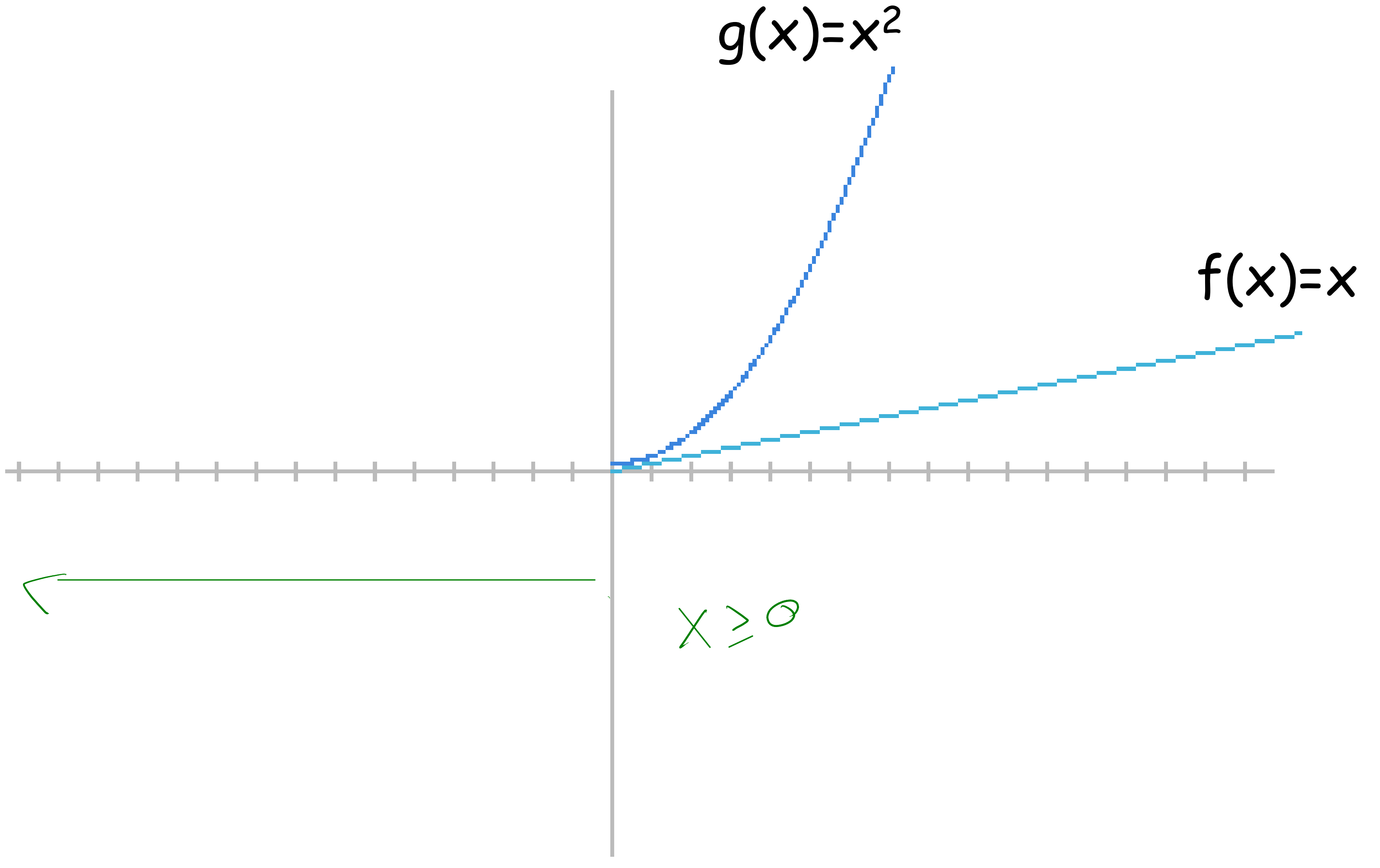
1157 day

3.17 [illegible]



n!
2n
n2
n log n
n
log n
1
16
8
4
2
1
1
2
3
4
5
nn

Crecimiento de funciones

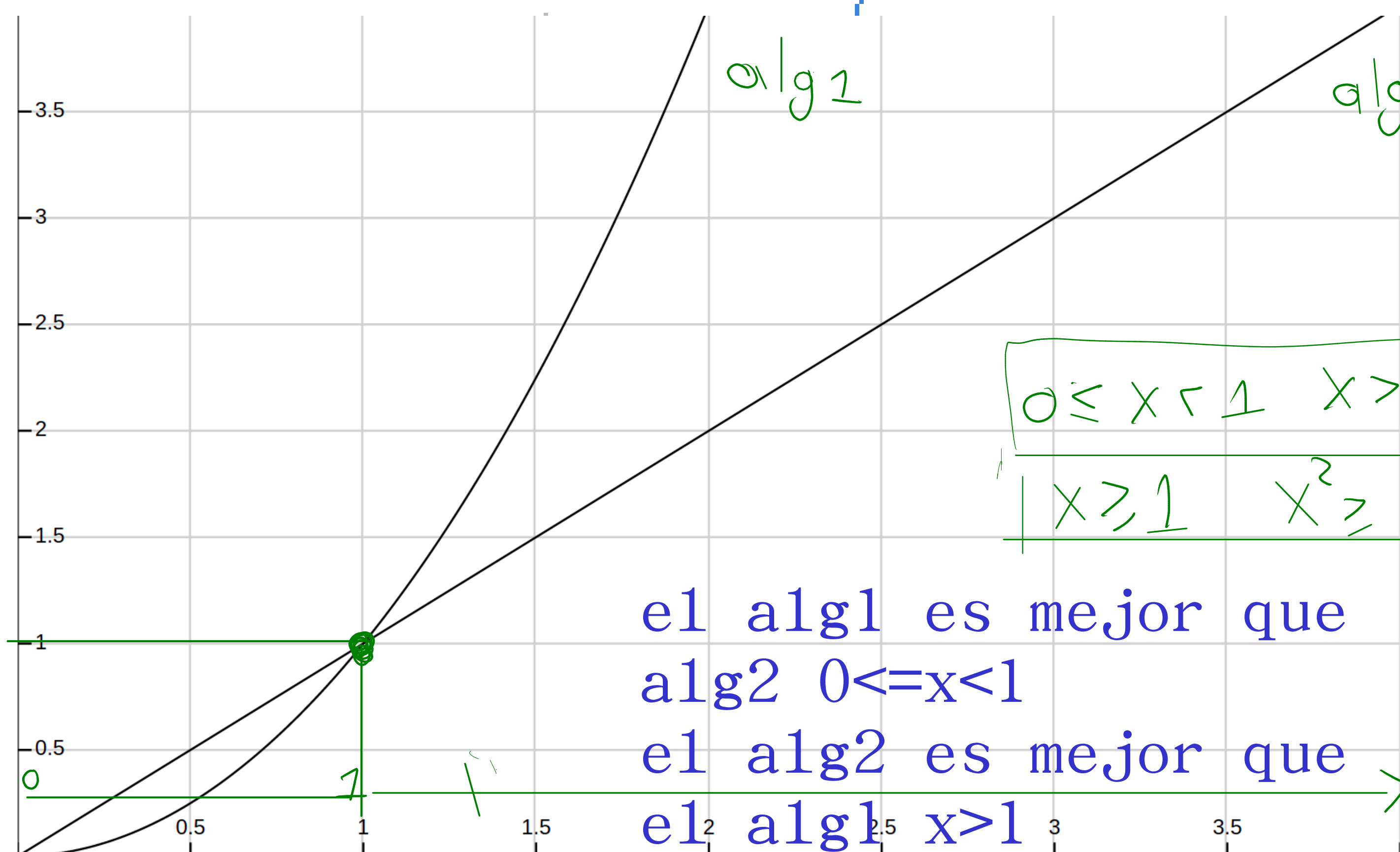


Crecimiento de funciones

¿Cuándo se cruzan?

$g(x)=x^2$

$f(x)=x$



$0 \le x < 1 \quad x > x^2$

$x \ge 1 \quad x^2 \ge x$

el alg1 es mejor que alg2 0<=x<1
el alg2 es mejor que el alg1 x>1

$\to \infty$

$x < 1$

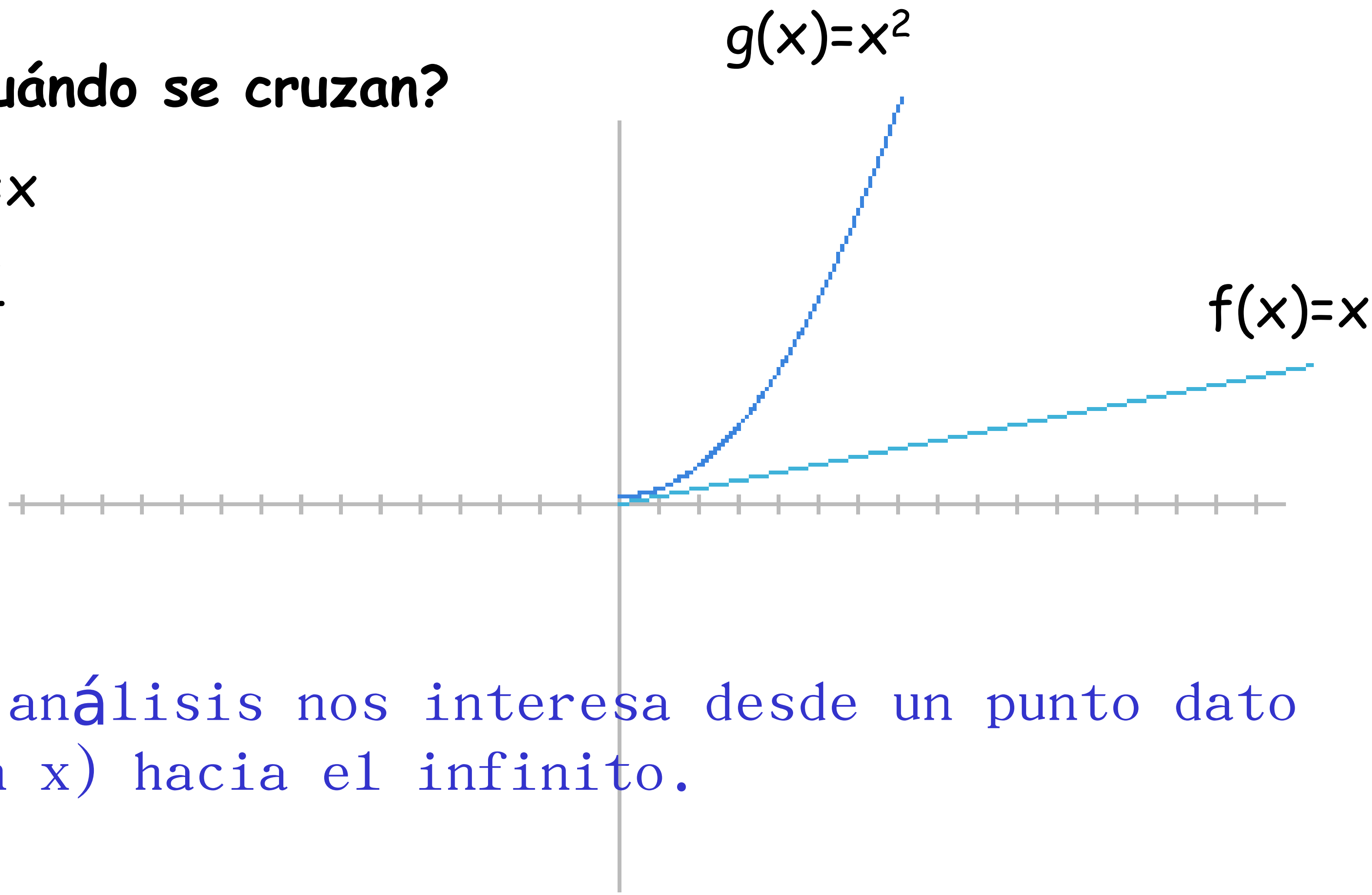
$x \ge 1$

Crecimiento de funciones

¿Cuándo se cruzan?

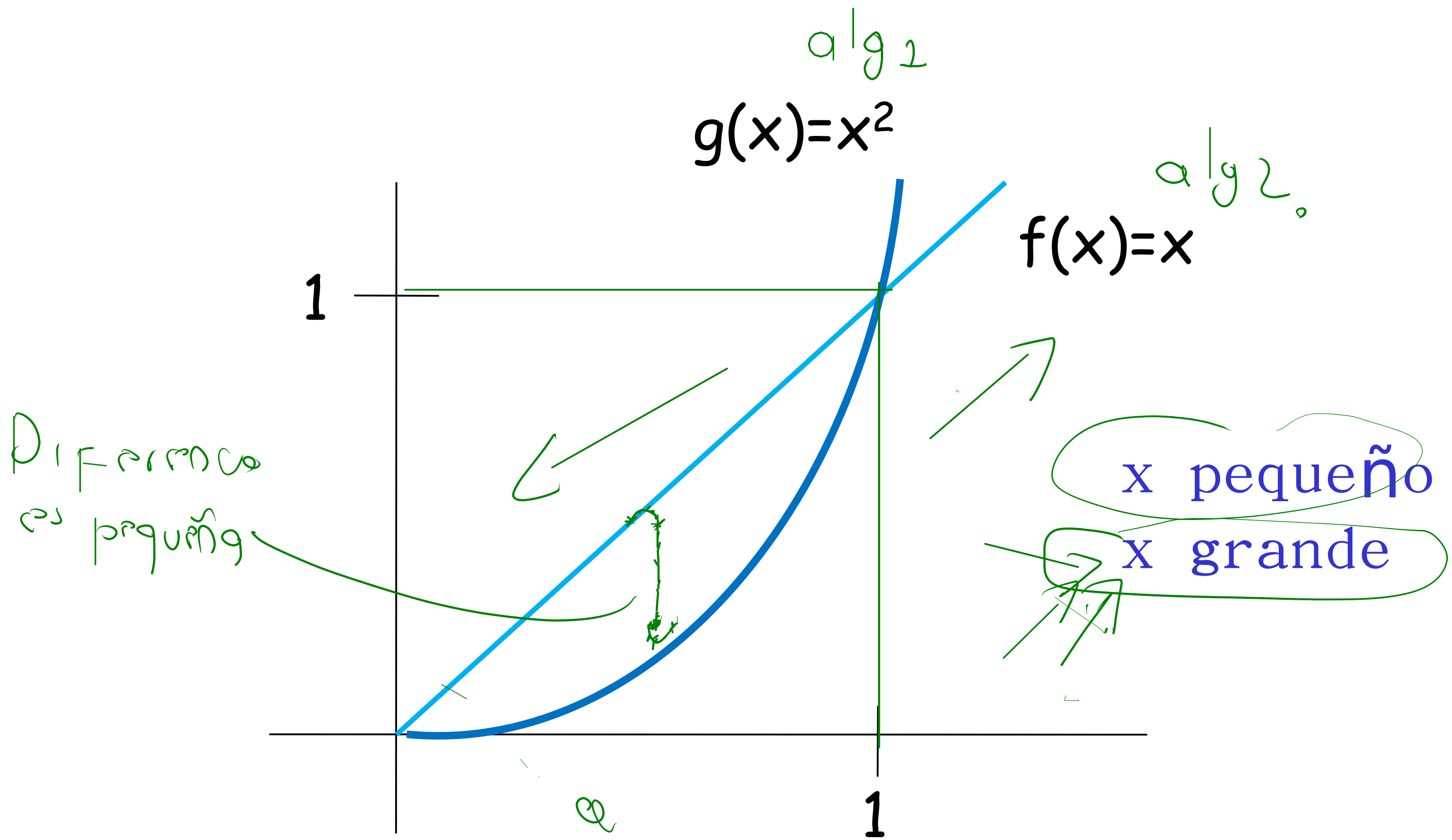
$x^2=x$

$x=1$



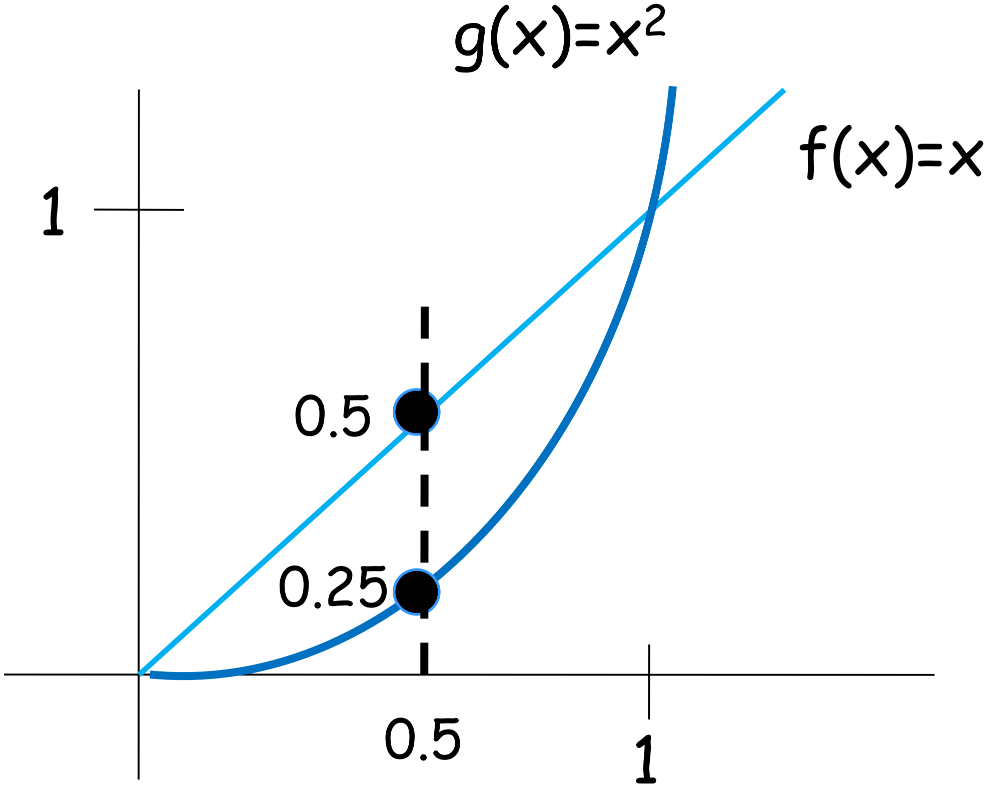
El análisis nos interesa desde un punto dato (un x) hacia el infinito.

Crecimiento de funciones

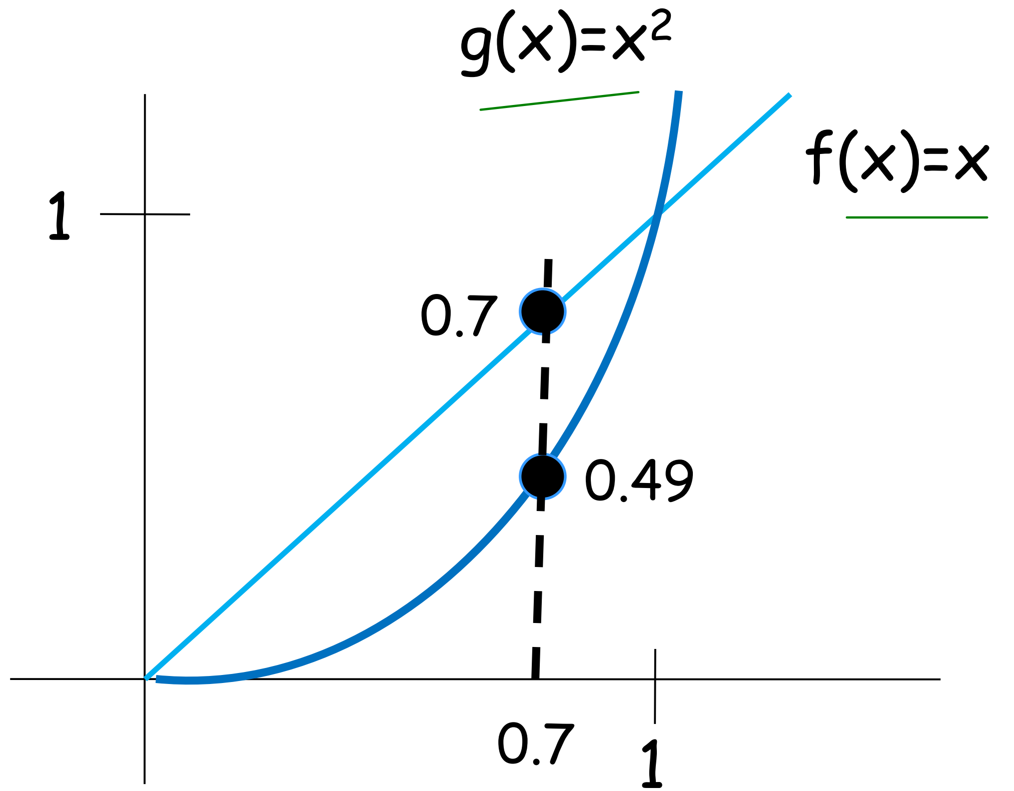


¿Para elegir un algoritmo? ¿Hacia donde me interesa mirar hacia los menores que 1 o los mayores que 1?

Crecimiento de funciones

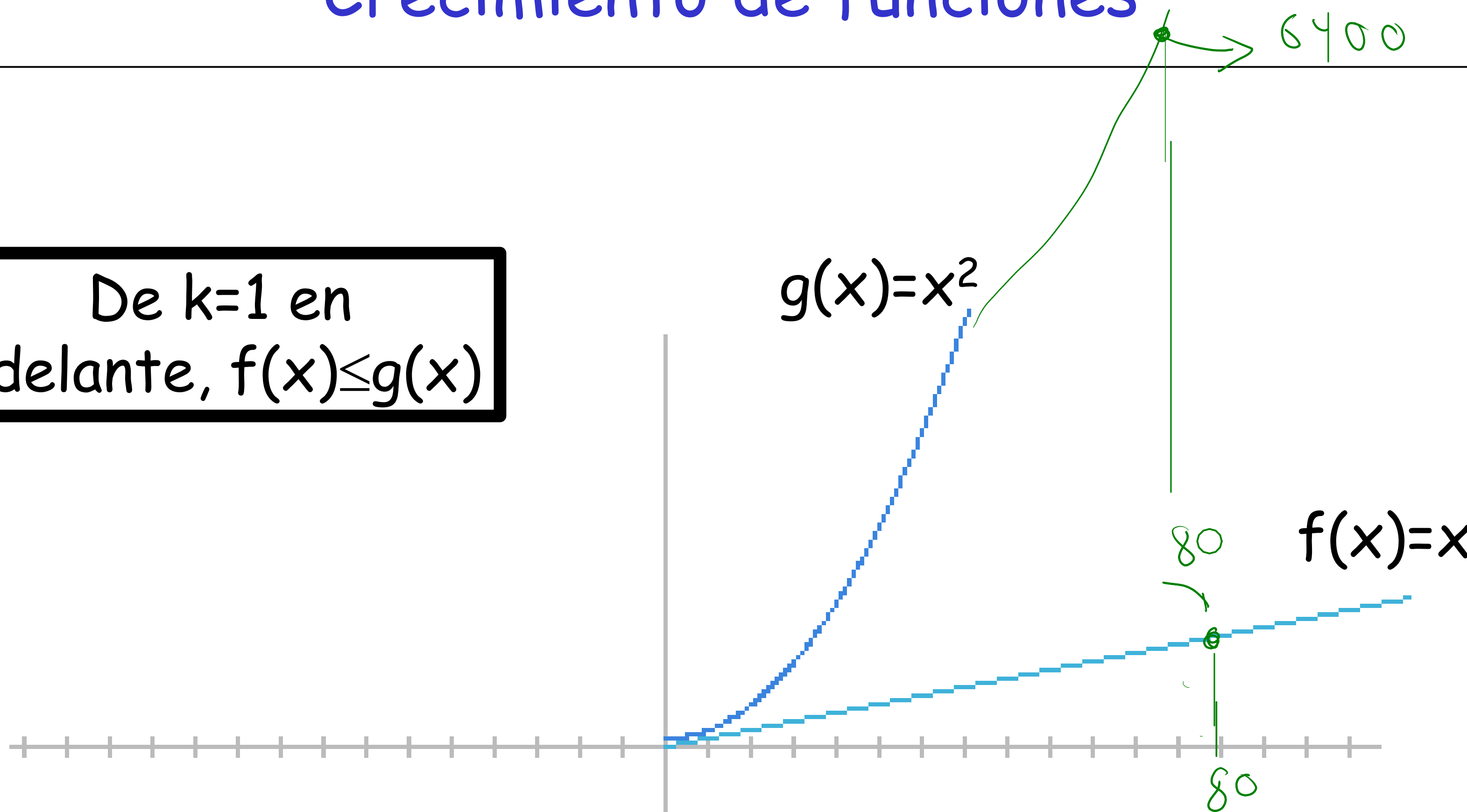


Crecimiento de funciones

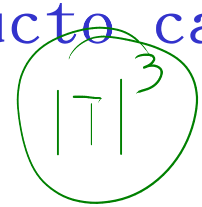


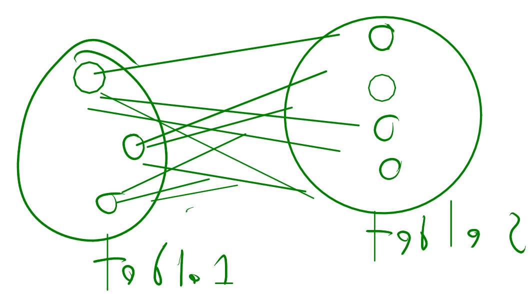
Crecimiento de funciones

De k=1 en adelante, $f(x) \leq g(x)$



JOIN <-- Producto cartesiano



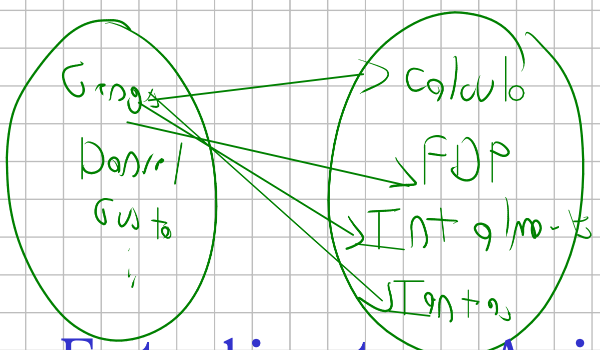


Cantidad de pasos = COMPLEJIDAD COMPUTACIONAL

Tabla de estudiantes

Tabla de asignaturas

Producto cartesiano ¿Que significa?



Estudiante Asignaturas

1) JOIN Estudiante y Asignaturas $|E||A|^2$

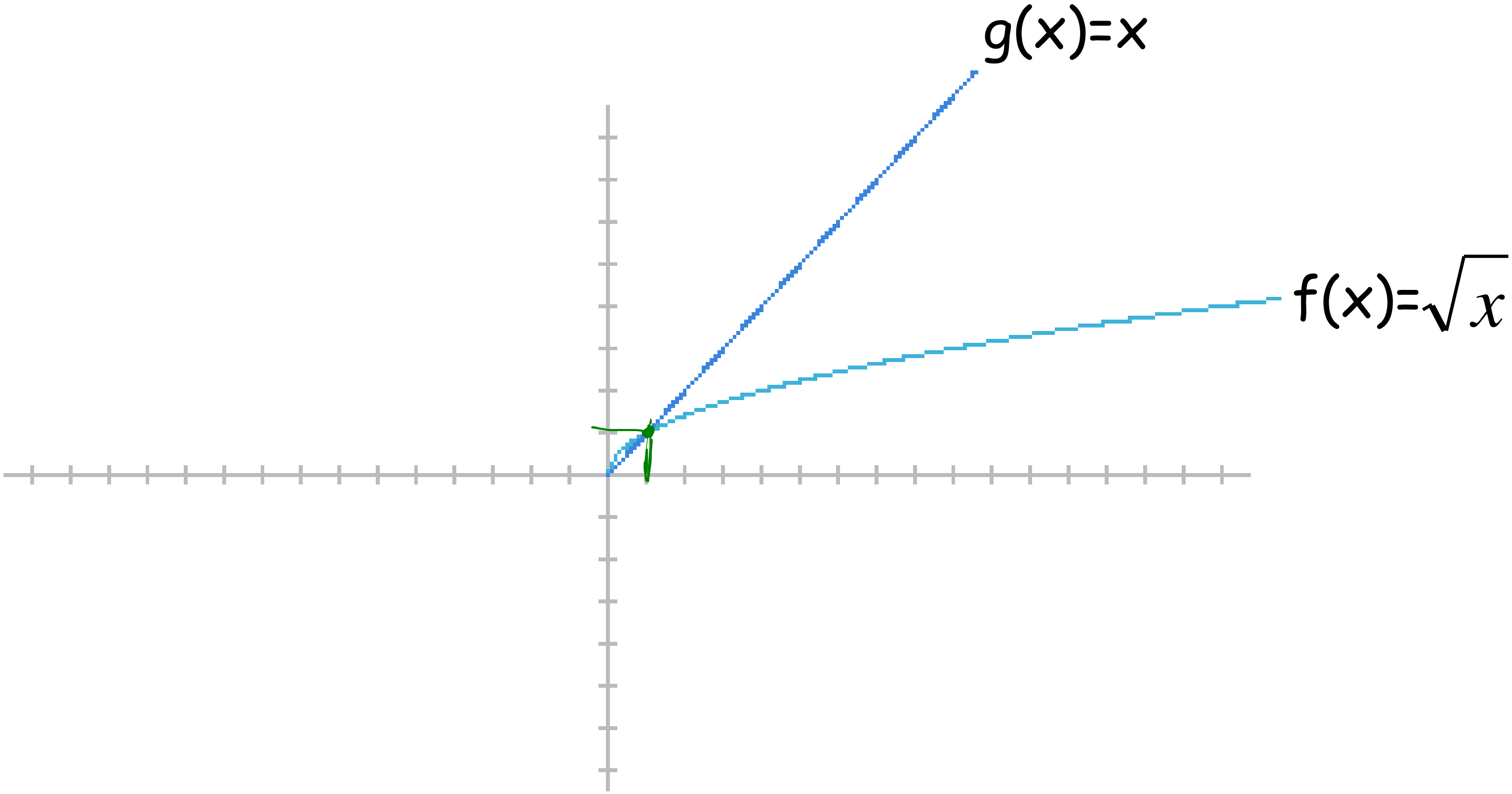
2) Filtro -> Las materias que ha visto el estudiante

500 sin filtrar, 50 filtrando

500^3 vs 50^3

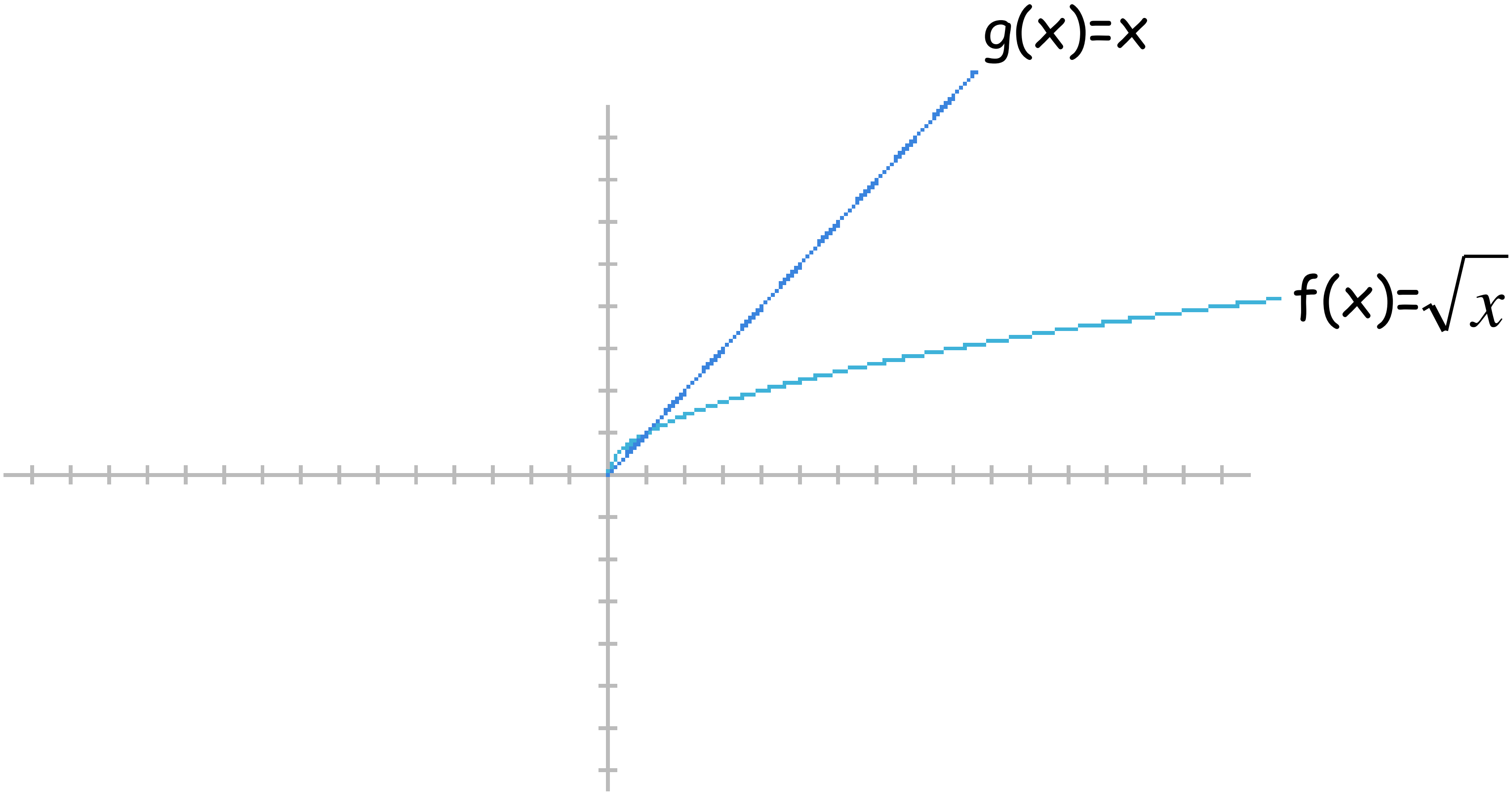
Crecimiento de funciones

Analice el crecimiento de las siguientes funciones



Crecimiento de funciones

¿Cuándo se cruzan?

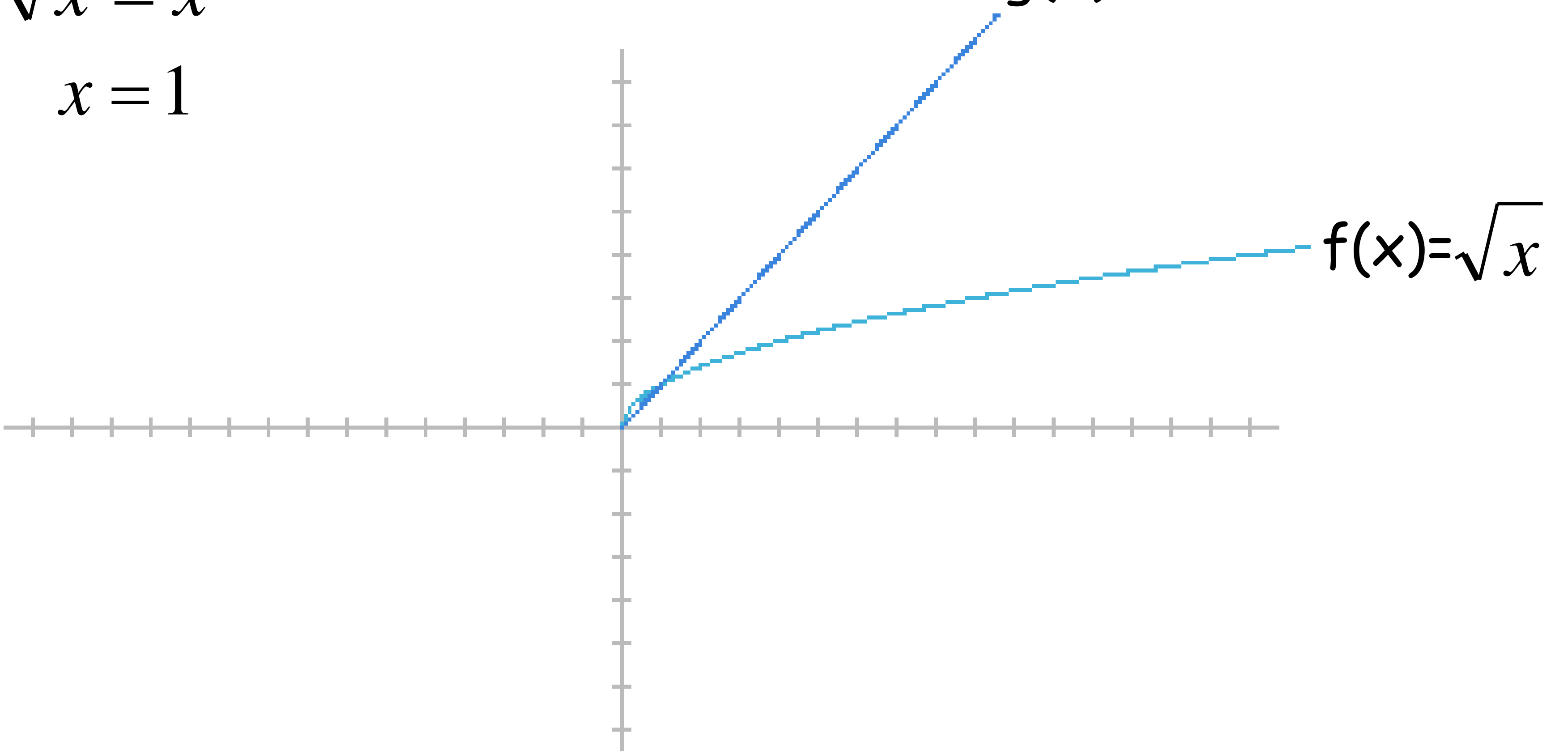


Crecimiento de funciones

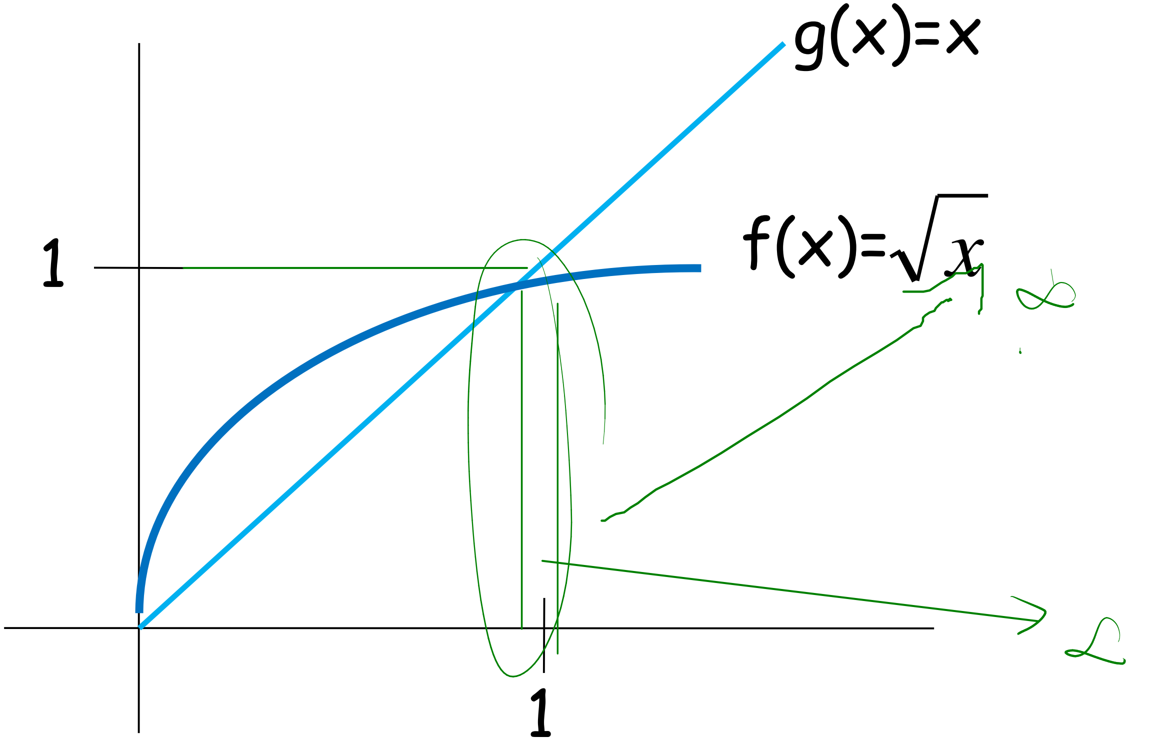
¿Cuándo se cruzan?

$$\sqrt{x} = x$$

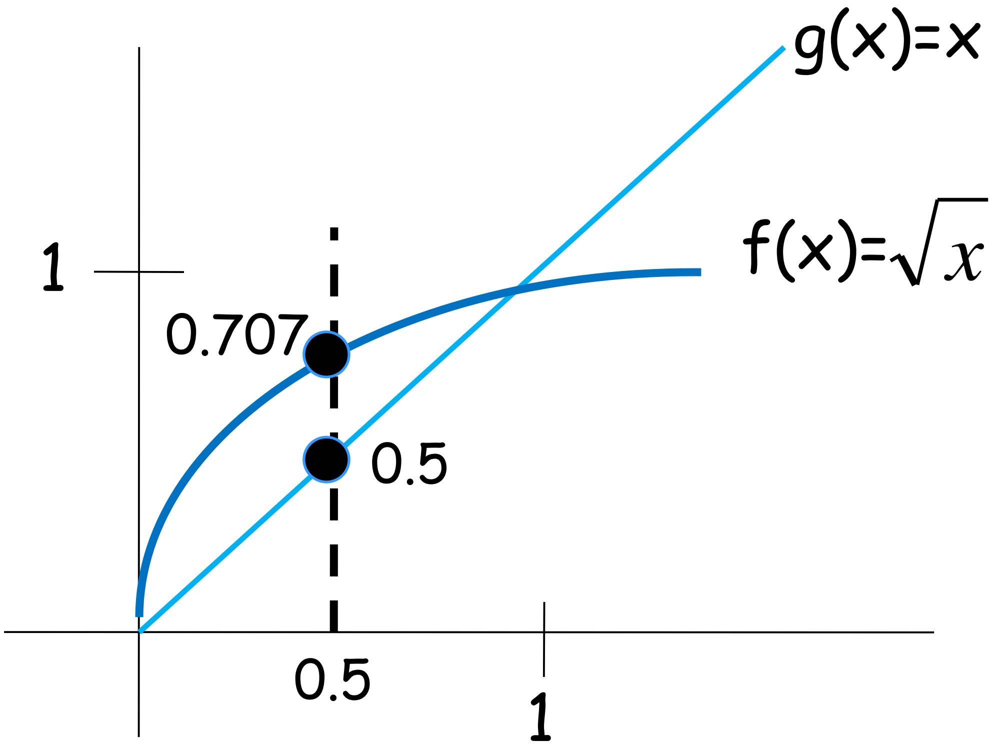
$$x = 1$$



Crecimiento de funciones

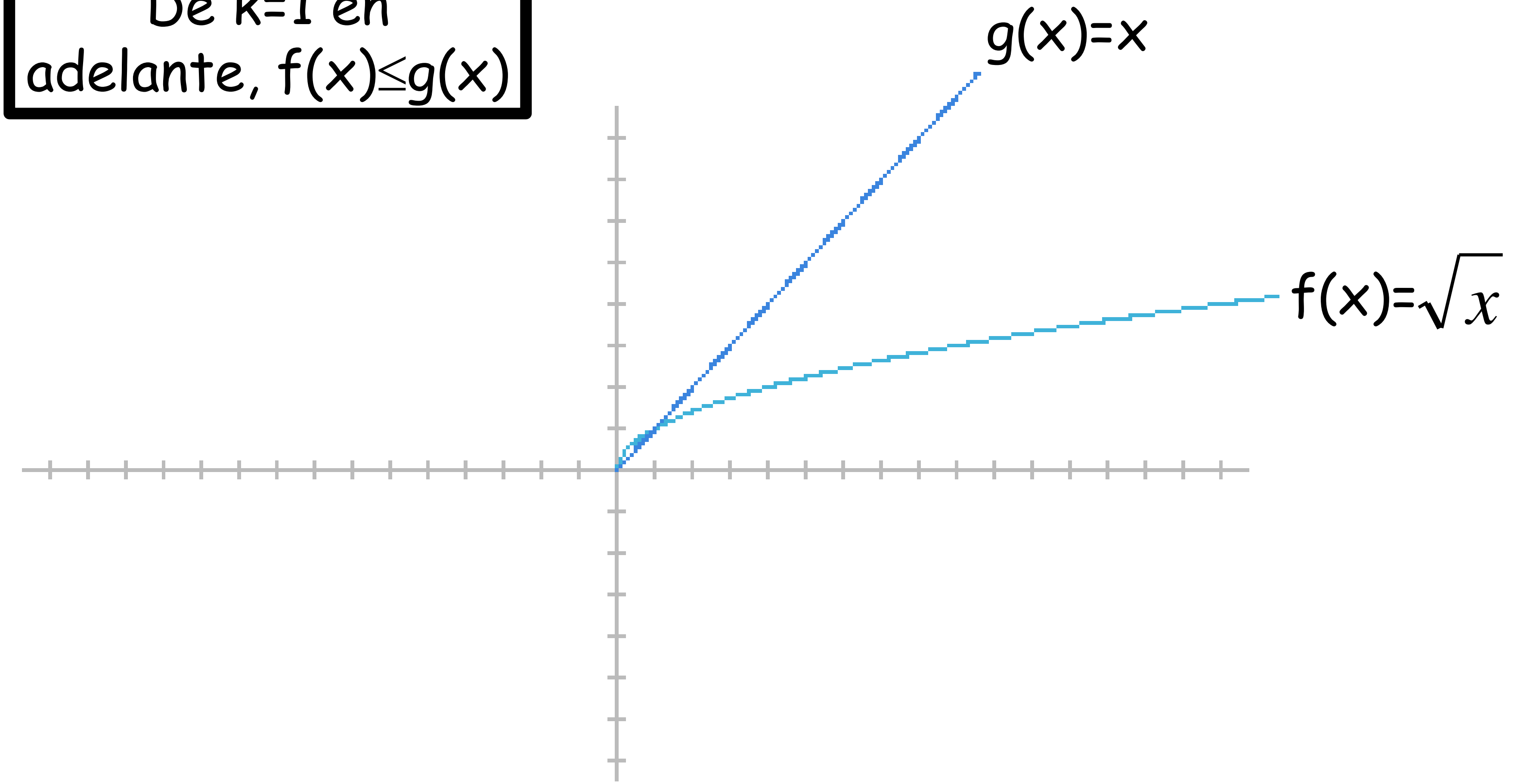


Crecimiento de funciones



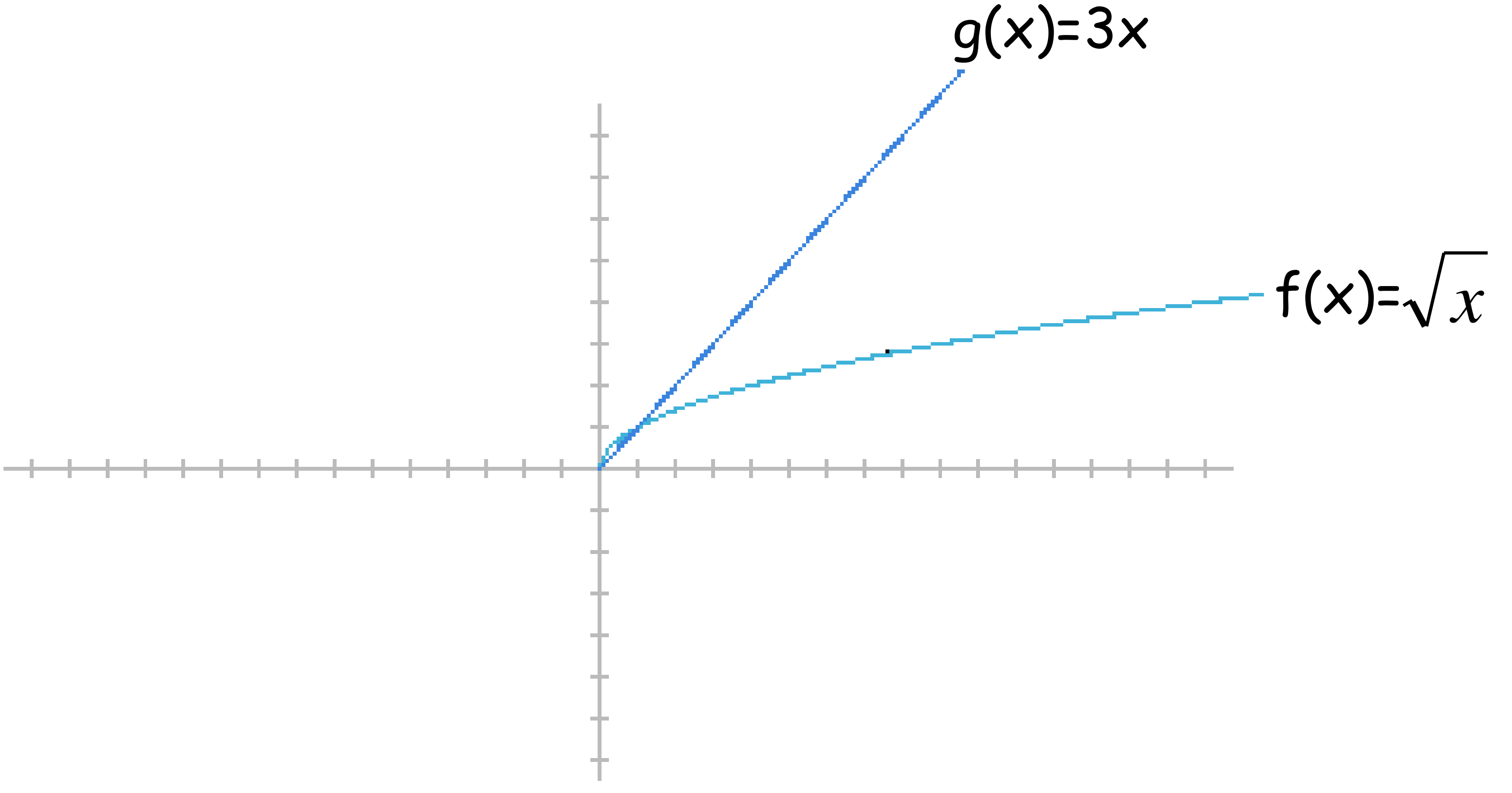
Crecimiento de funciones

De k=1 en adelante, $f(x) \leq g(x)$



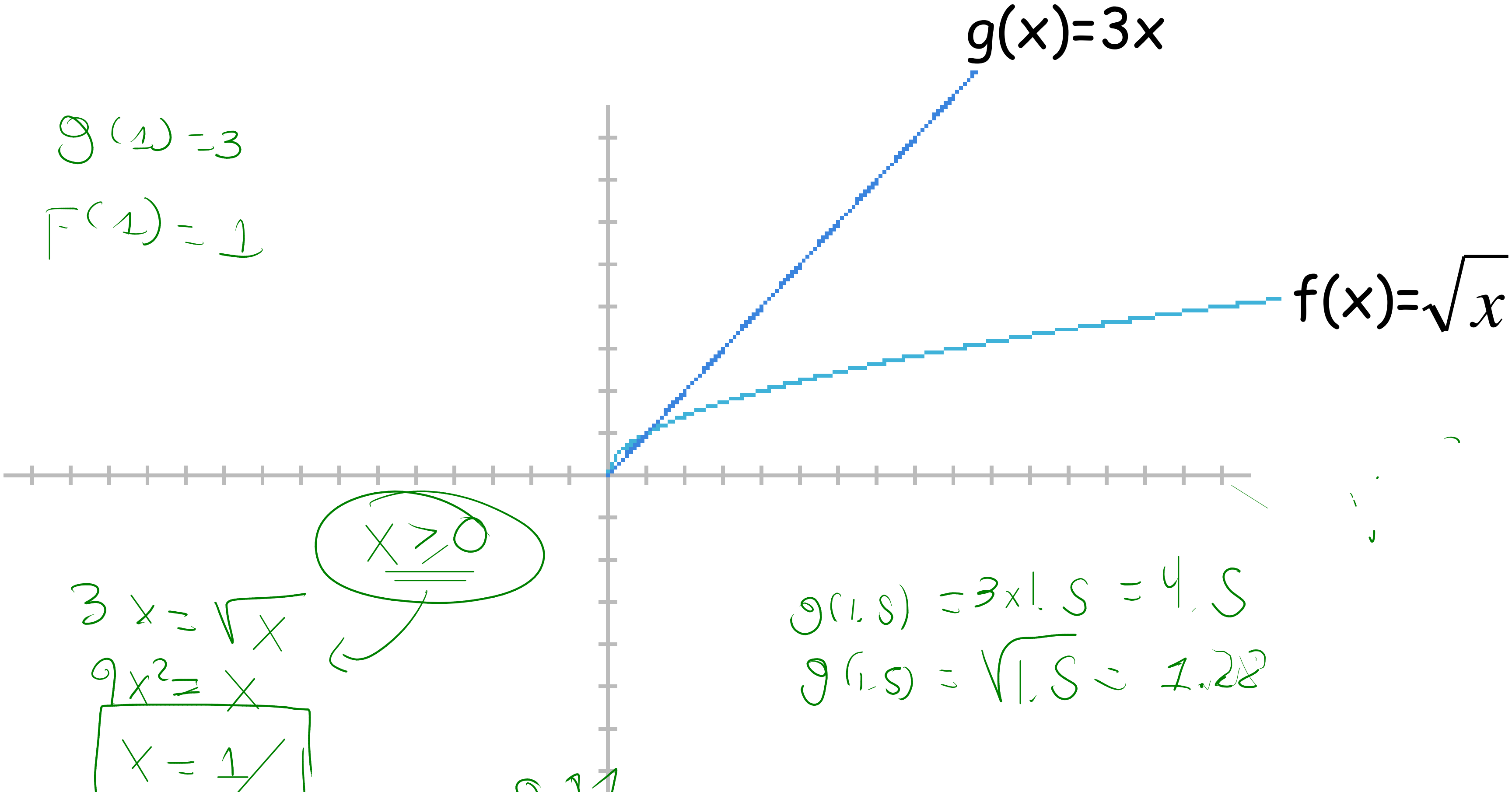
Crecimiento de funciones

Analice el crecimiento de las siguientes funciones



Crecimiento de funciones

¿Cuándo se cruzan?



$g(1) = 3$

$f(1) = 1$

$x \geq 0$

$3x = \sqrt{x}$

$9x^2 = x$

$\boxed{x = \frac{1}{9}}$

$0,11$

$g(1.5) = 3 \times 1.5 = 4.5$

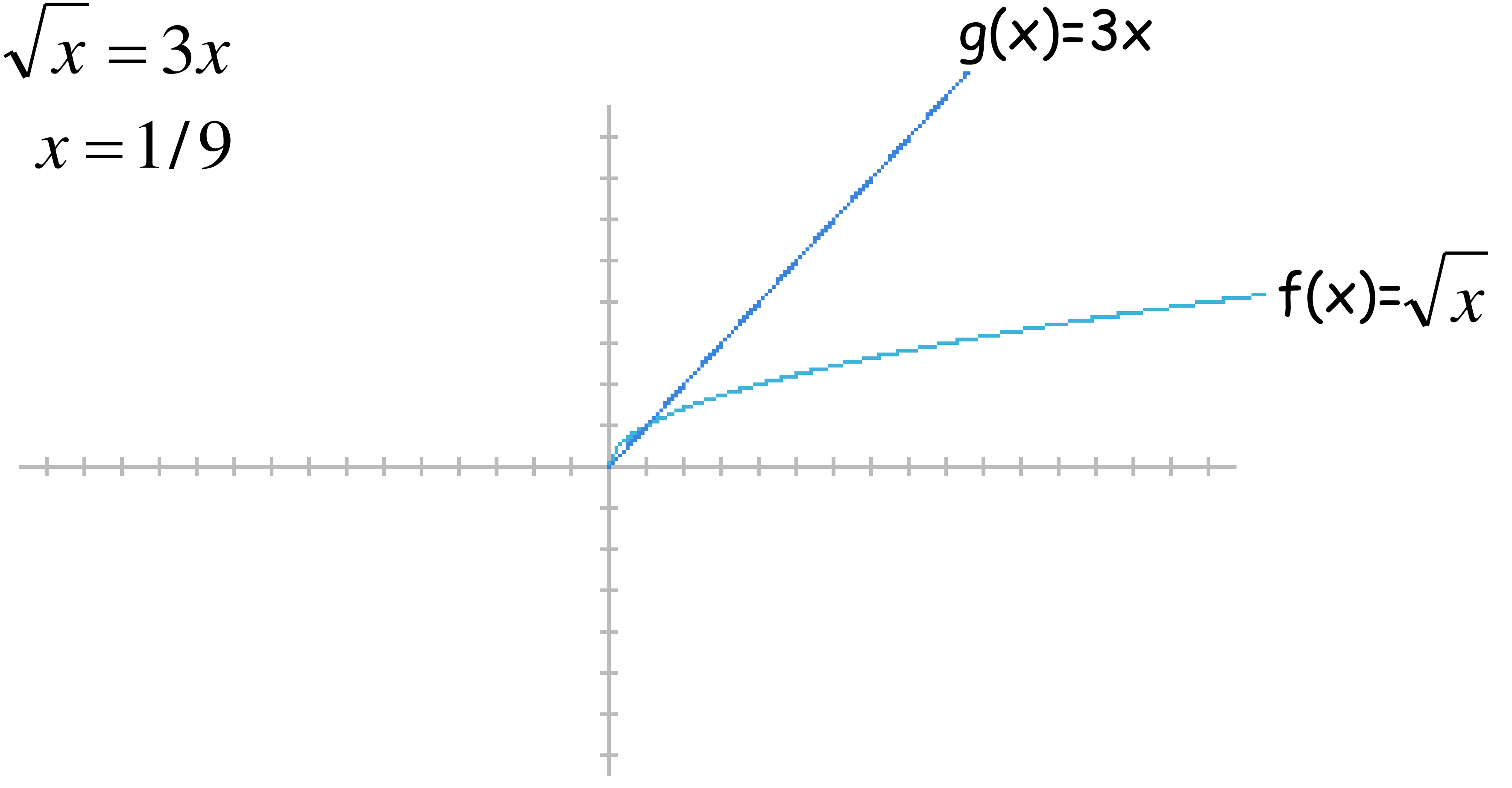
$g(1.5) = \sqrt{1.5} \simeq 1.22$

Crecimiento de funciones

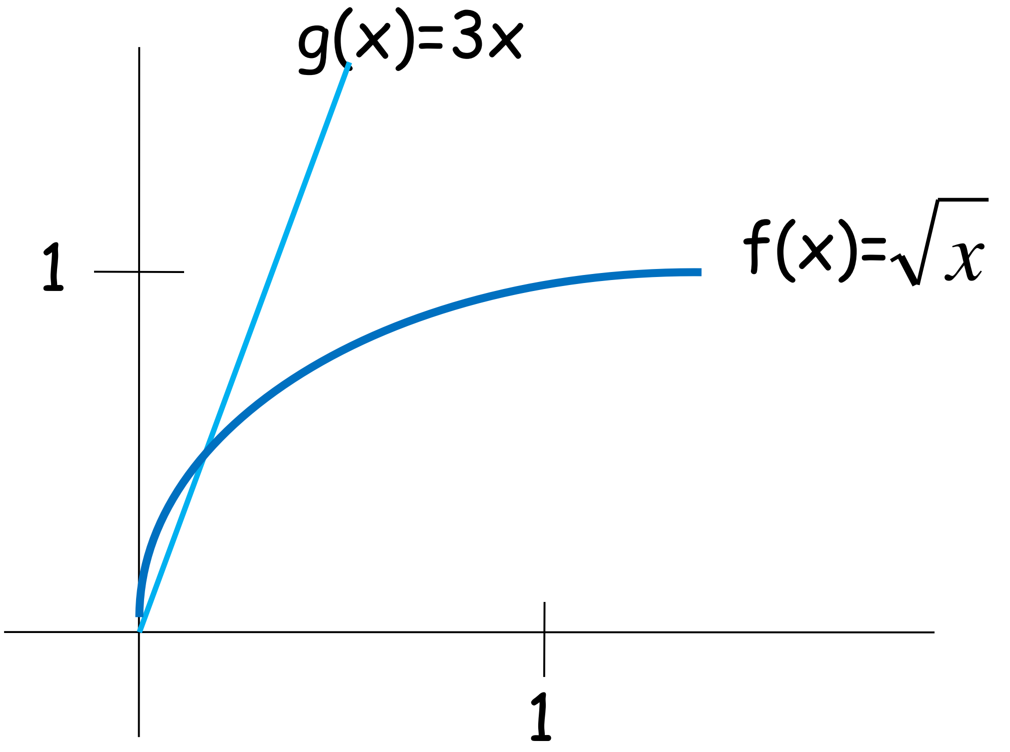
¿Cuándo se cruzan?

$$\sqrt{x} = 3x$$

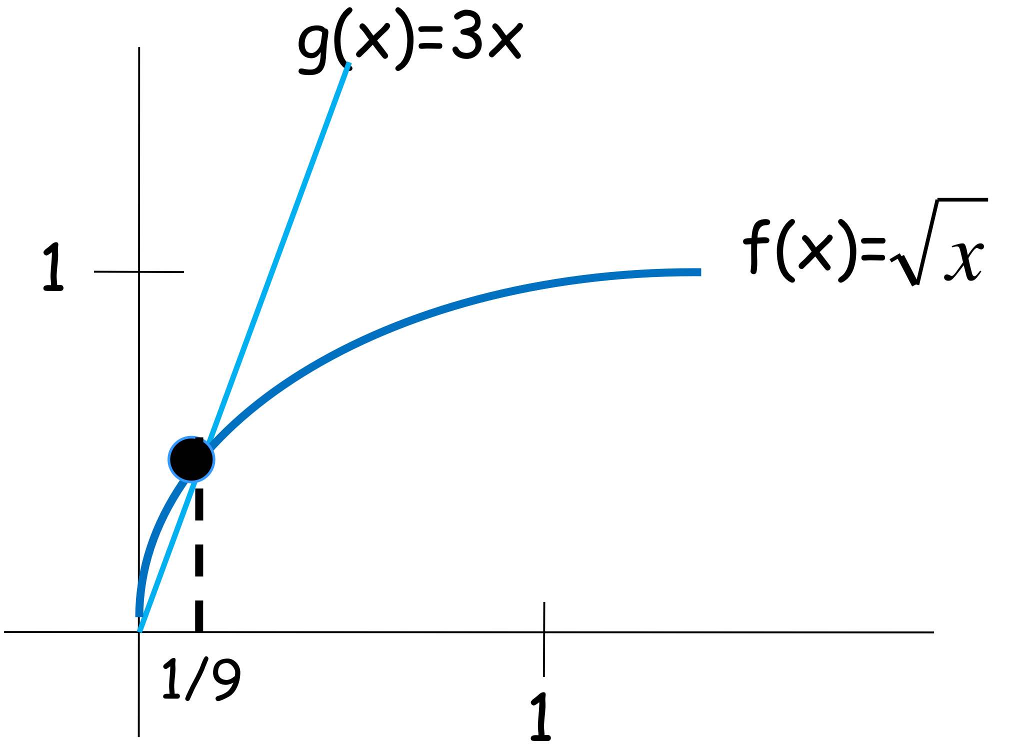
$$x = 1/9$$



Crecimiento de funciones

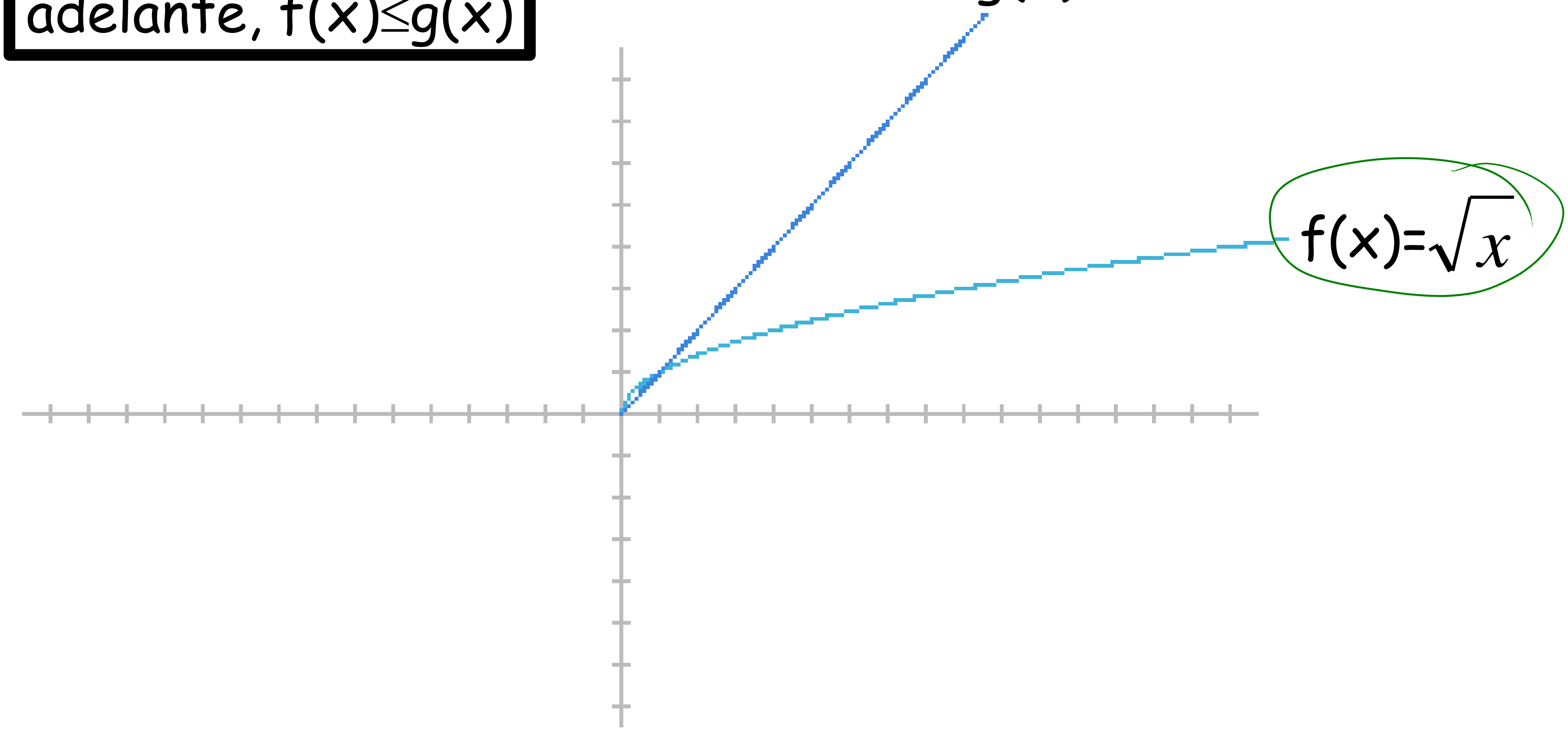


Crecimiento de funciones



Crecimiento de funciones

De $k=1/9$ en adelante, $f(x) \leq g(x)$

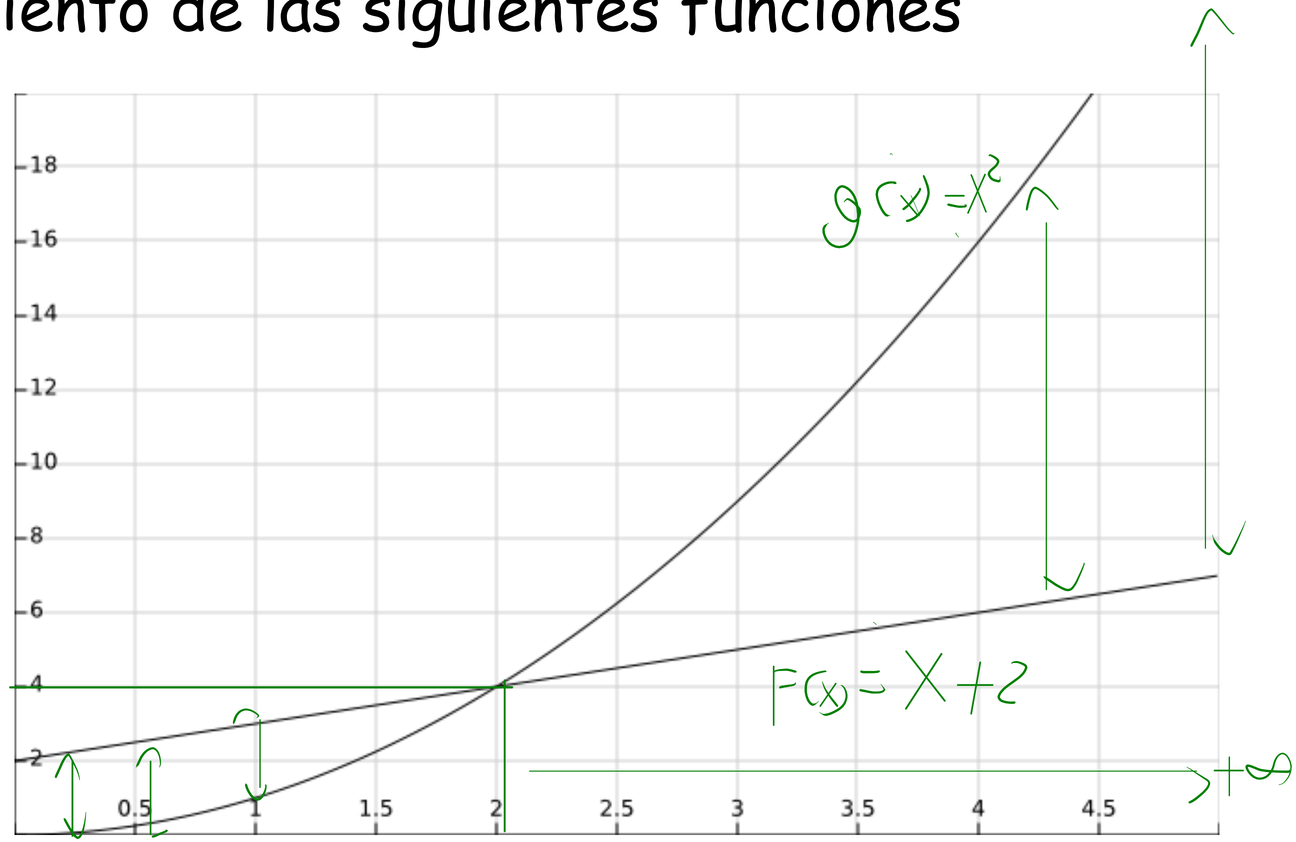


Crecimiento de funciones

Analice el crecimiento de las siguientes funciones

$g(x)=x^2$

$f(x)=x+2$



¿Cuando f(x) es mejor que g(x)?

$\boxed{X \geq 2}$

Crecimiento de funciones

Analice el crecimiento de las siguientes funciones

$g(x)=x^2$

$f(x)=x+2$

De k=2 en adelante, $f(x) \leq g(x)$

Crecimiento de funciones

Analice el crecimiento de las siguientes funciones

$g(x)=x^2$

$f(x)=2x+24$

¿Cual de las dos es mejor y porque?

F(x) es mejor que g(x) $\boxed{x \geq 6}$

$x^2 = 2x + 24$

$x^2 - 2x - 24 = 0 \qquad ax^2 + bx + c$

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \qquad \frac{2 \pm \sqrt{4 - 4(1)(-24)}}{2(1)}$$

$$\frac{2 \pm \sqrt{4 + 96}}{2} \qquad \frac{2 \pm \sqrt{100}}{2} \qquad \frac{2 \pm 10}{2}$$

$$\frac{2+10}{2} = \frac{12}{2} = 6$$

$$\cancel{\frac{2-10}{2} = \frac{-8}{2} = -4}$$

Crecimiento de funciones

Analice el crecimiento de las siguientes funciones

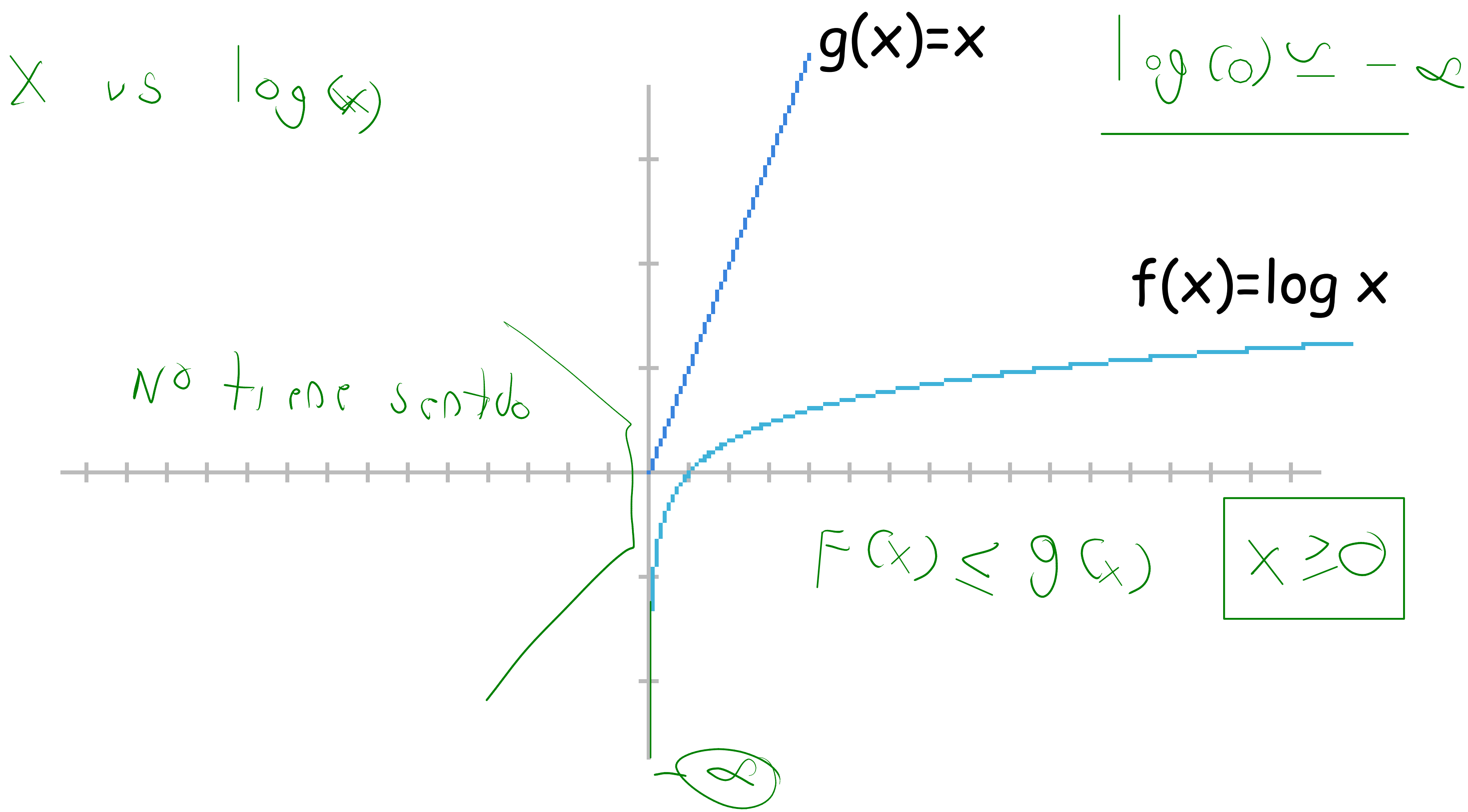
$g(x)=x^2$

$f(x)=2x+24$

$\boxed{\text{De } k=6 \text{ en adelante, } f(x) \leq g(x)}$

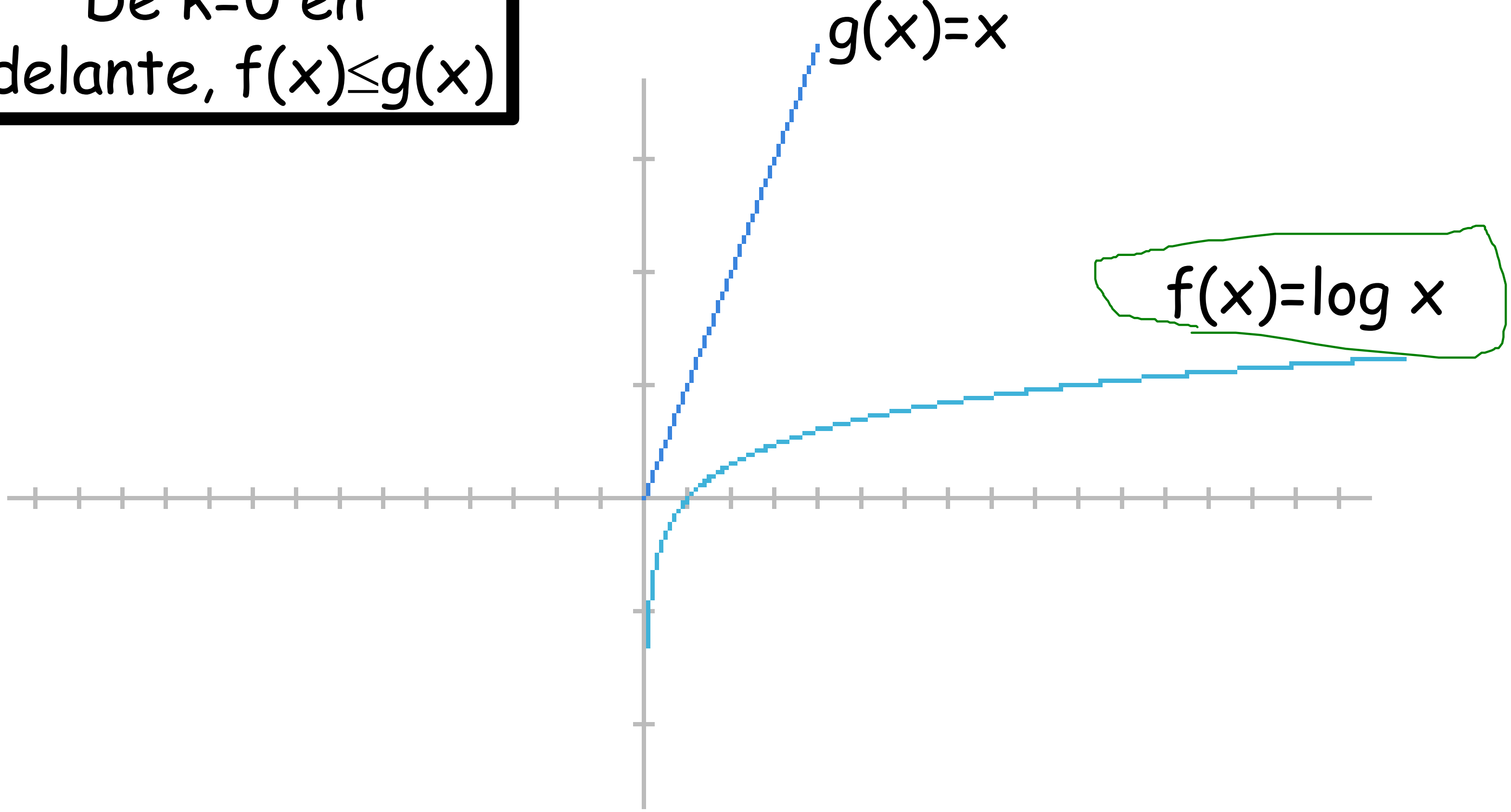
Crecimiento de funciones

Analice el crecimiento de las siguientes funciones



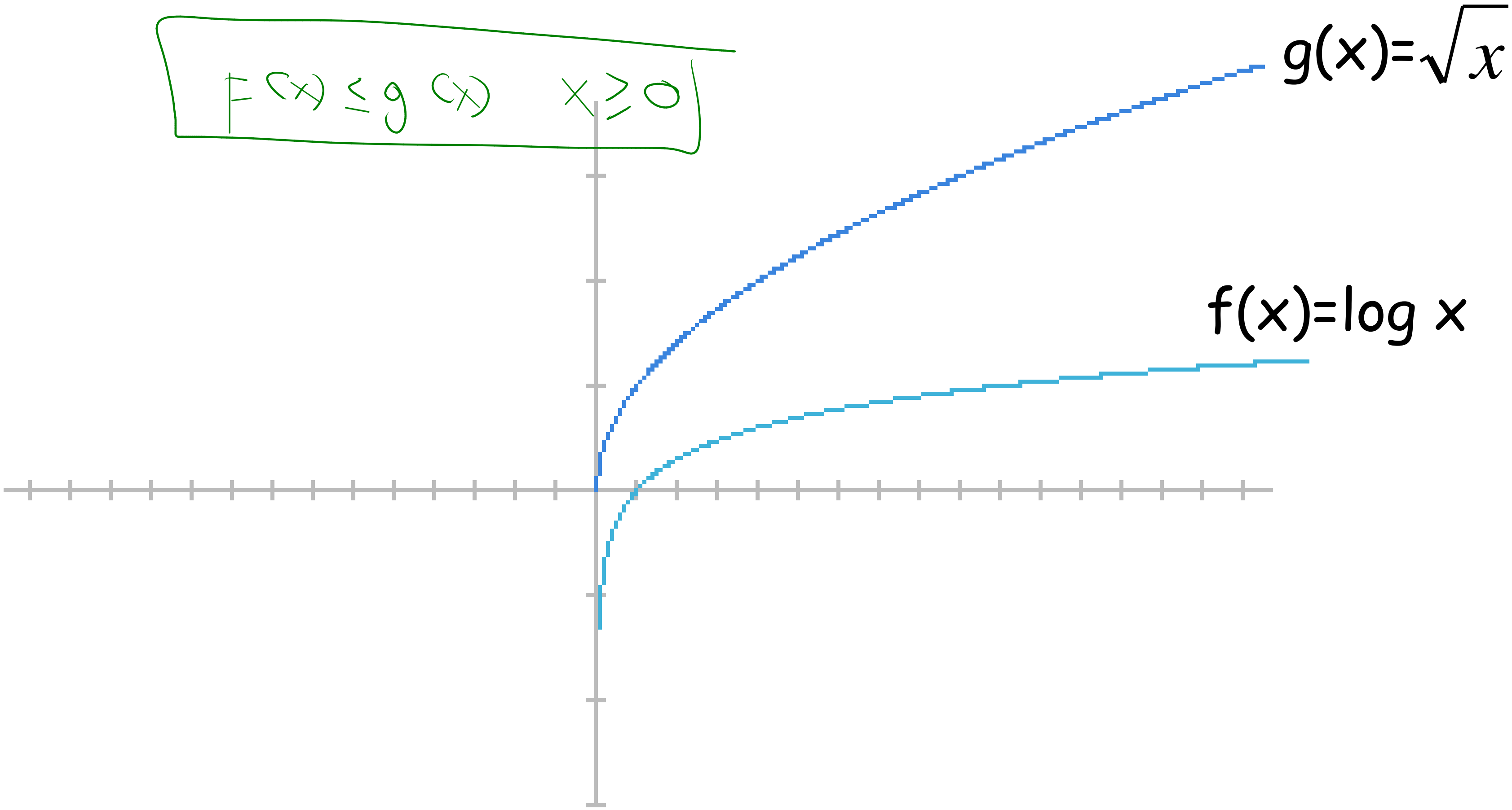
Crecimiento de funciones

De k=0 en adelante, f(x)≤g(x)



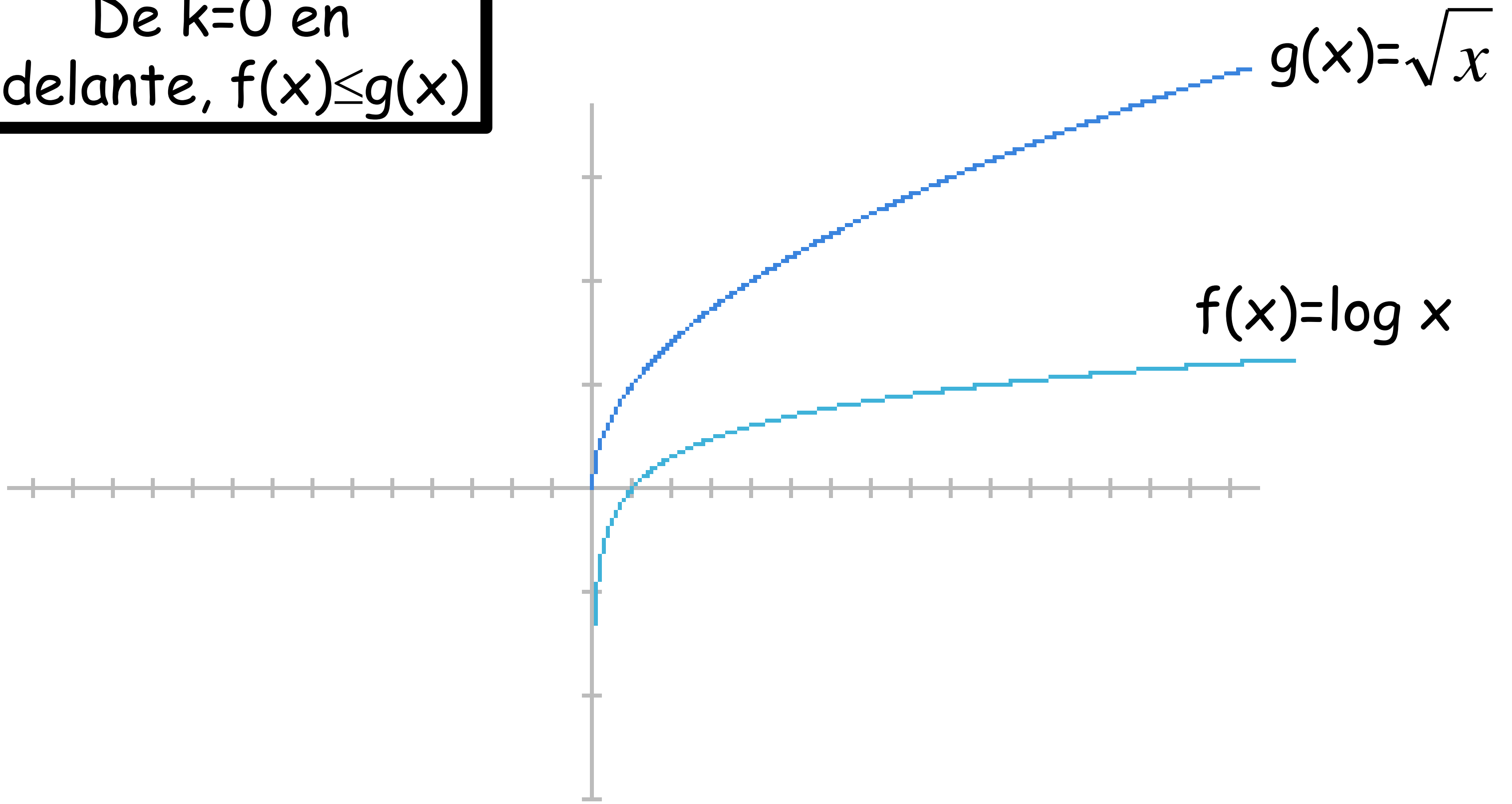
Crecimiento de funciones

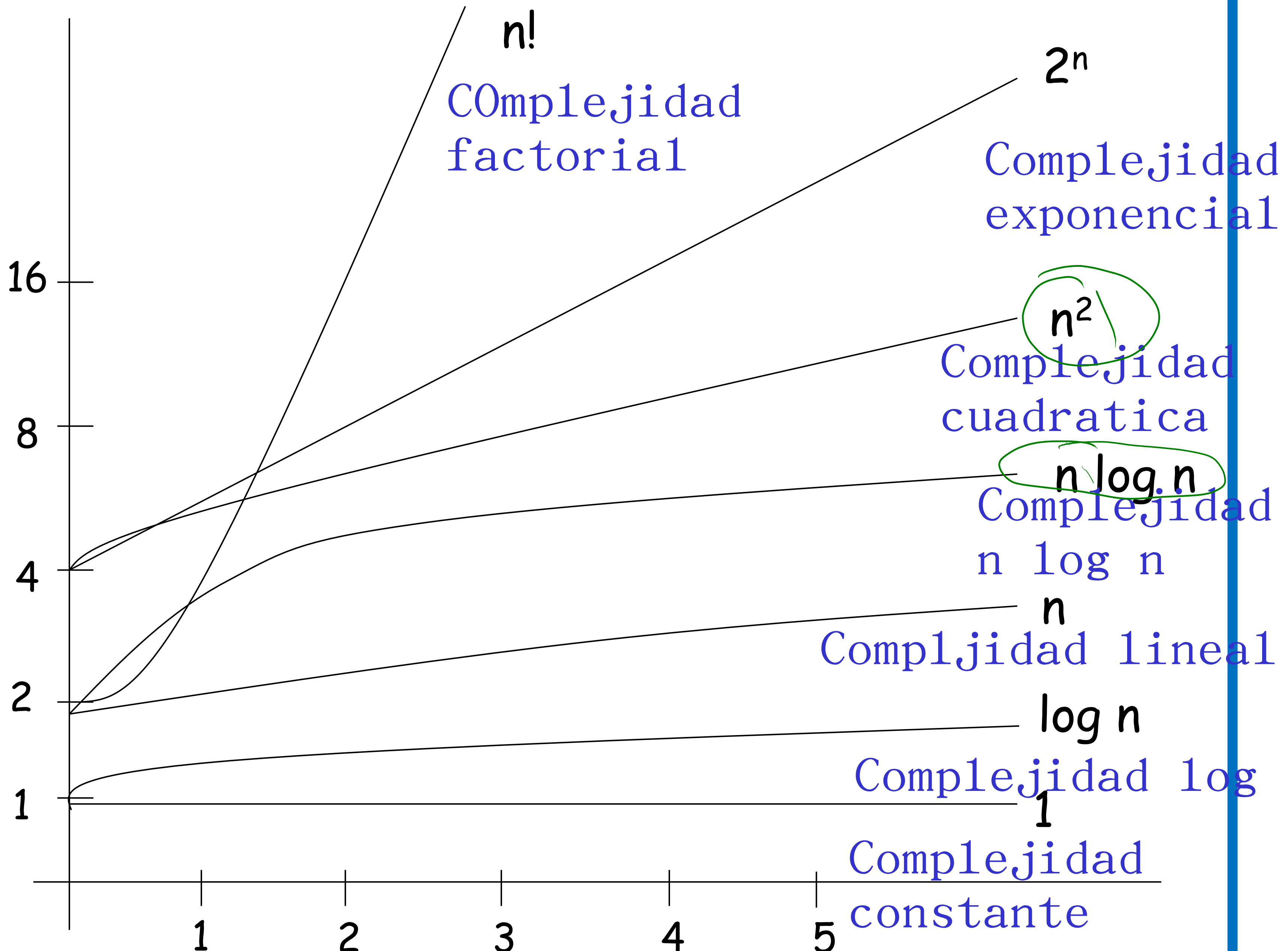
Analice el crecimiento de las siguientes funciones



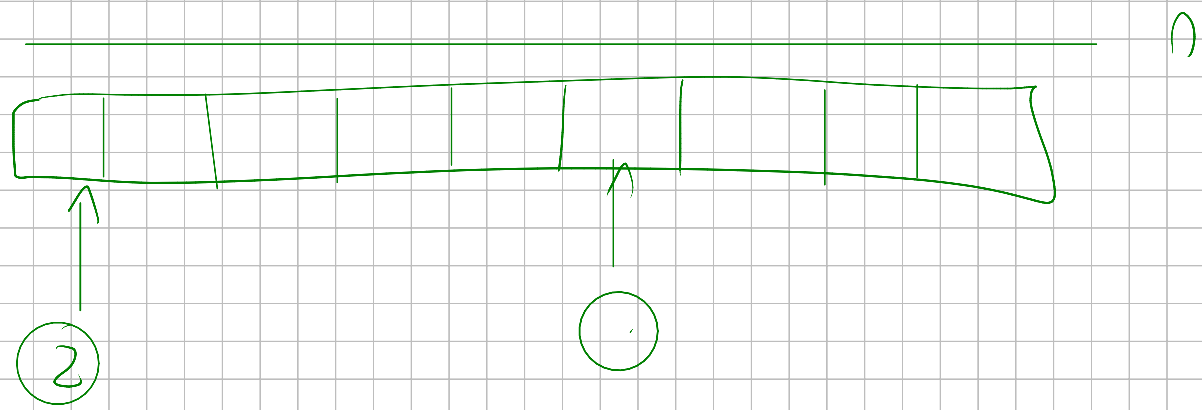
Crecimiento de funciones

De k=0 en adelante, $f(x) \leq g(x)$




n!
COmplejidad factorial
2^n
Complejidad exponencial
n^2
Complejidad cuadratica
n log n
Complejidad n log n
n
Compljidad lineal
log n
Complejidad log
1
Complejidad constante
16
8
4
2
1
1
2
3
4
5

Pensar en un problema sobre un arrreglo de tamaño n, cuya complejidad sea 1.



Tomar el valor de una posición cualquiera del arreglo toma tiempo 1

Crecimiento de funciones

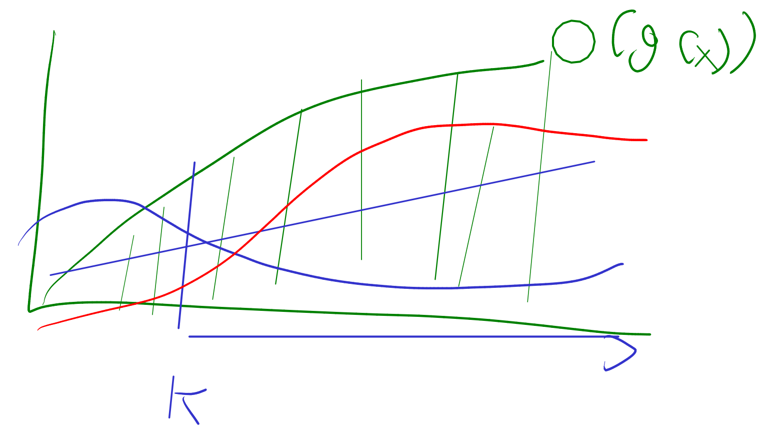
Notación O

PEOR CASO, techo de lo que espero del algoritmo.

Sean f y g dos funciones, se dice que $f(x)=O(g(X))$ si se cumple que

$$f(x) \leq g(x)$$

para $x>k$



Crecimiento de funciones

Muestre que $7x^2=O(x^3)$

Crecimiento de funciones

Muestre que $7x^2=O(x^3)$

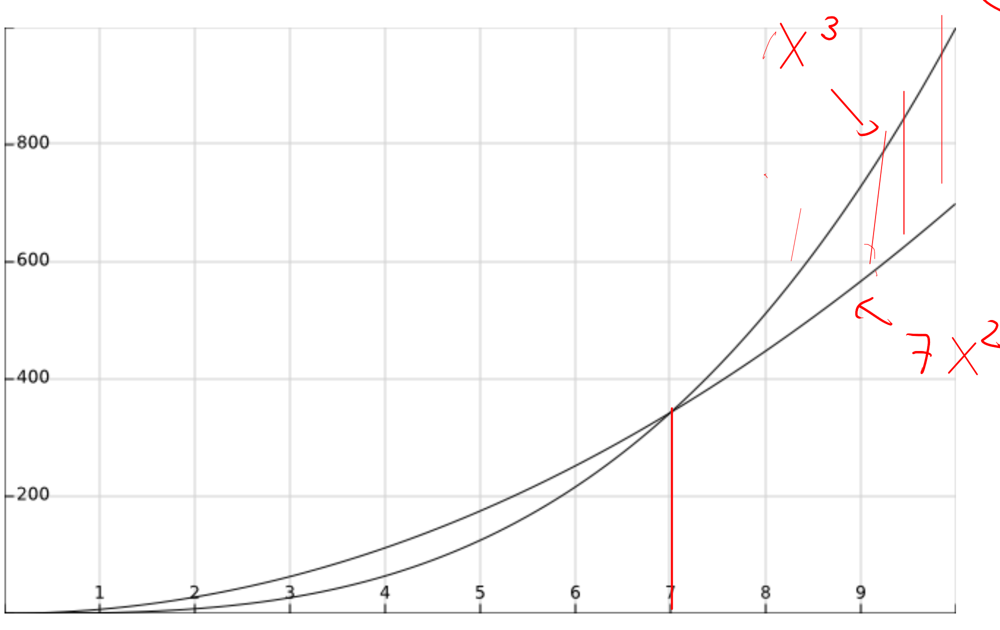
$7x^2 < x^3$, para $x>7$

$\boxed{x > 7}$

$7x^2 < x^3$

$\frac{7x^2}{x^2} < \frac{x^3}{x}$

$7 < x$



$7x^2$ es $O(x^3)$

$x > 7$

Crecimiento de funciones

Muestre que $2x\text{-}1=O(x^2)$

$f(x)$ es $O(g(x))$ sii

$x \geq k \quad f(x) \leq g(x)$

$\longrightarrow \infty$

$2x - 1 \leq x^2$

$$\boxed{x^2 - 2x + 1 \geq 0}$$

$$\frac{2 \pm \sqrt{4 - 4(1)}}{2(1)} = \frac{2 \pm \sqrt{0}}{2} = 1$$

$$\boxed{x \geq 1}$$

Crecimiento de funciones

Muestre que $2x-1=O(x^2)$

$2x-1 < x^2$, para $x>1$

Crecimiento de funciones

Muestre que $3x+10=O(x^2)$

Crecimiento de funciones

Muestre que $3x+10=O(x^2)$

$3x+10 < x^2$, para $x>5$

Crecimiento de funciones

Muestre el valor de k para el cual se cumple cada una de las siguientes relaciones:

- $6x+16=O(x^2)$
- $\sqrt{6x} = O(3x)$

Crecimiento de funciones

Aplicación

techo

La notación O permite establecer una $\boxed{\text{cota superior}}$ al tiempo dado por un algoritmo

Crecimiento de funciones

Aplicación

La notación O permite establecer una cota superior al tiempo dado por un algoritmo

- Suponga que tiene dos algoritmos cuyos tiempos de ejecución están acotados de la siguiente forma:
  - $T_1(n) = O(n^2)$
  - $\boxed{T_2(n) = O(\log n)}$ ← mejor tiempo

- ¿Cuál algoritmo escogería?

Crecimiento de funciones

Aplicación

La notación O permite establecer una cota superior al tiempo dado por un algoritmo

• Suponga que tiene dos algoritmos cuyos tiempos de ejecución están acotados de la siguiente forma:

- $T_1(n) = O(n^2)$
- $T_2(n) = O(\log n)$

• ¿Cuál algoritmo escogería? **El algoritmo 2**

El análisis de la cota superior se hace de la siguiente manera

$$f(x) \text{ es } O(g(x))$$

$$f(x) \leq C * g(x) \qquad \boxed{x \geq k}$$

$$\boxed{C \geq 1}$$

$$x+2 \text{ es } O(x^2) \qquad \boxed{x \geq \frac{1}{2}}$$

$$x+2 \leq C * x^2 \qquad C = 4$$

$$x+2 \leq 4 * x^2$$

$$\boxed{0 \leq 4x^2 - x - 2} \quad 4x^2 - x - 2 \geq 0$$

$$\frac{1 \pm \sqrt{1 + 2 \times 4}}{2(4)} \qquad \frac{1 \pm \sqrt{9}}{8} = \frac{1 \pm 3}{8}$$

$$\frac{4}{8} = \frac{1}{2} \qquad \frac{-2}{8} = \frac{-1}{4}$$

$6x^2 + 3$ es $O(x^2)$

$6x^2 + 3 \leq c \cdot x^2$

$\boxed{c \geq 1}$

$6x^2 + 3 \leq 9x^2$

$0 \leq 3x^2 - 3$

$3 \leq 3x^2$

$1 \leq x^2$

$\boxed{x \geq 1}$

$1 \leq x$

mayor grado

$8x^2 + 6x + 3$ es $O(x^2)$

$8x^2 + 6x + 3 \leq c \cdot x^2$

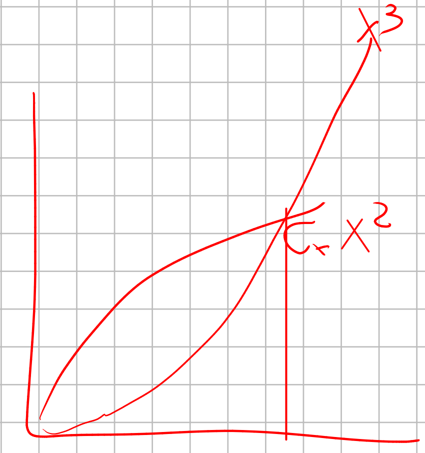
$\boxed{c \geq 1}$

$8x^2 + 6x + 3 \leq 10x^2$

$2x^2 - 6x - 3 \geq 0$

$x \geq 3 + 2\sqrt{3}$

$$8x^3 + 6x^2 - 100x - 10000 \text{ es } O(x^2)$$



$$8x^3 + 6x^2 - 100x - 10000 \le c_x x^2$$

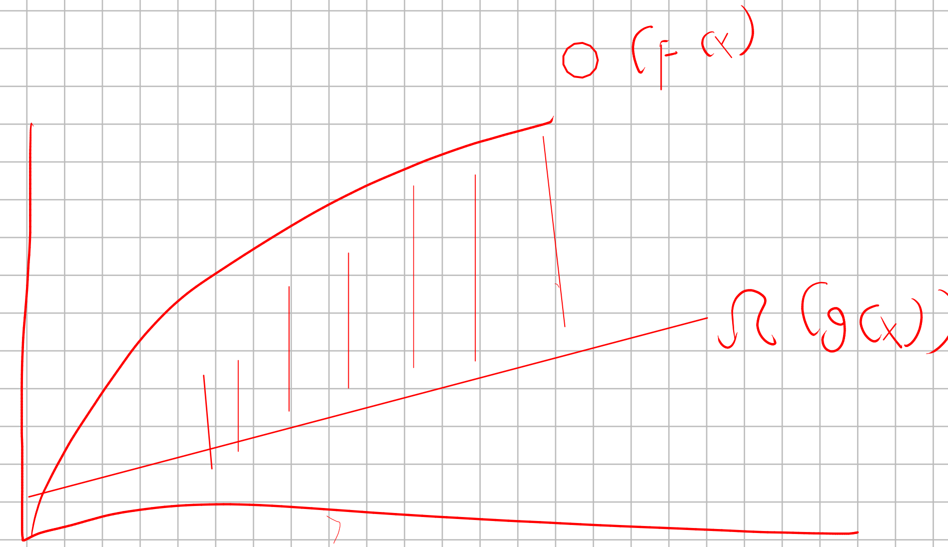
$$8x^3 + 6x^2 - 100x - 10000 \le 10^8 x^2$$

$$\boxed{0 \le -8x^3 + (10^8 - 6)x^2 + 100x + 10000}$$

NO EXISTE en los $\mathbb{R}$

NO es $O(x^2)$

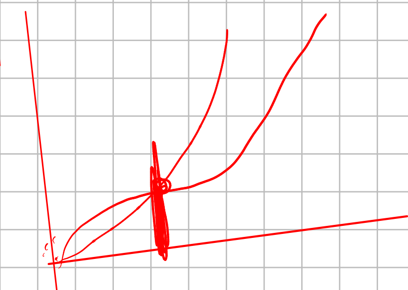
Notación piso Ω



f(x) es Ω(x) si

$$\boxed{f(x) \ge C \times g(x) \quad x \ge k}$$

$$\boxed{x \ge \frac{3}{2}}$$



$6x^2$ es $\Omega(x)$

$$6x^2 \ge C \times x$$

$$6x^2 \ge 9x$$

$$\boxed{6x^2 - 9x \ge 0}$$

$$x(6x-9) = 0$$

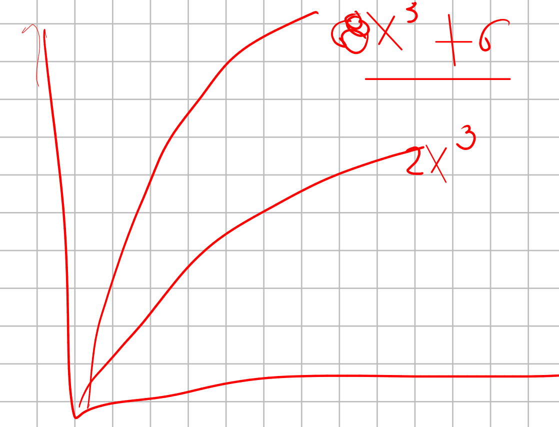
$x = 0$

$$\boxed{x = \frac{3}{2}}$$

$8x^3 + 6 \quad \Omega(x^3)$

$8x^3 + 6 \geq C \cdot x^3$

$x \geq k$



$8x^3 + 6 \geq 2x^3$

$6x^3 + 6 \geq 0$

$x \geq 0$

$x^3 \geq -\frac{6}{6}$

$\boxed{x^3 \geq -1}$

$F(x)$ es $O(g(x))$

PEOR CASO

Que g(x) es una COTA SUPERIOR (TECHO) de f(x)

$$f(x) \text{ es } O(g(x)) \text{ Sii } f(x) \leq c * g(x), x \geq k$$

El c dele un valor que le convenga, y luego demuestre que existe un k que cumple la relación. Truco, deje la ecuación c*g(x)-f(x) >= 0

$f(x)$ es $\Omega(h(x))$

$$f(x) \geq c * h(x), x \geq k, c \geq 1$$

h(x) es COTA INFERIOR (PISO) de f(x), que f(x) nunca va ser menor que h(x), x>=k

Escoja un c cualquier(CONVENIENTE) y busca el valor de k que hace cumplir el teorema

MEJOR CASO

$f(x)$ es $\Theta(g(x))$ — $f(x)$ es $O(g(x))$; $f(x)$ es $\Omega(g(x))$

COTA AJUSTADA

Truco: Suele funcionar en funciones con el mismo grado polinomico

$8x^2+6$ es $\Theta(x^2)$ — $8x^2+6$ es $O(x^2)$ $\boxed{x \geq \sqrt{3}}$; $8x^2+6$ es $\Omega(x^2)$ $\boxed{x \geq 1}$

$O(x^2)$

$8x^2+6 \leq C \cdot x^2$ $\quad x \geq k \quad C \geq 1$

$8x^2+6 \leq 10 \cdot x^2$ $\quad 10x^2-8x^2-6 \geq 0$

$2x^2-6 \geq 0 \quad 2x^2 \geq 6$

$x^2 \geq 3$

$x \geq \sqrt{3}$

$\Omega(x^2)$

$8x^2+6 \geq C \cdot x^2$ $\quad \boxed{C \geq 1} \quad C=2$

$8x^2+6 \geq 2x^2$ $\quad 8x^2+6-2x^2 \geq 0$

$6x^2+6 \geq 0 \quad 6x^2 \geq 6 \quad x^2 \geq 1$ $\quad \boxed{x \geq 1}$