For i 🡨 2 to quar O(sqrt(n))

for (int i = 1; i < n; i \*= 2) //O(log n)

for I 🡨 a to b

O(b)

for I 🡨 sqrt(b) to b

for I 🡨 sqrt(b)/ to b

for I 🡨 n to 1

for j 🡨 1 to i/2

fn

if n < 17

return 9;

return f(n -4)

g(n)

if n < -2

ruturn 2

return g(n-2) +g(n-5)

O(3^n)

g2(n)

if n < -2

ruturn 2

return g(n-2) +g(n-5) – g(n-4)

O(log3^n) = O(logn)

for (int I = 1; I < n; I = i\*3 +1)

sqrt(n)

for(int I = sqrt(n); I >= 1; I --)

n – sqrt(n)

for(int I = sqrt(n)

O(1)

for(int j = 3 \*I -3; j <= 3\*I; j++)

quicksort 具体细节怎么排的