

What does $O(< expr >)$ mean?

1

What does $\Theta(< expr >)$ mean?

2

What does $\Omega(< expr >)$ mean?

3

Say that the input represents a positive integer, x , what is the size of n ?

4

What does it mean by $O(1)$?

5

What is the minimum time for any sorting algorithm that uses only number comparisons.

6

How do you argue correctness of Euclid's algorithm?

7

What would half the correctness proof be for Euclid's algorithm?

8

The complexity (i.e. space/running time) has the complexity proportional to $\langle \text{expr} \rangle$.

The complexity (i.e. running time/space) is bounded by the $\langle \text{expr} \rangle$.

2

1

$\lfloor \log_b x \rfloor + 1$ Where b is the number representation, usually binary (so 2).

The complexity (i.e. running time/space) is at least by the $\langle \text{expr} \rangle$.

4

3

$n \log_2 n$

It takes a constant time, no matter the amount of data, to perform the operation.

6

5

As $r = a \bmod b$, $\exists q$ such that $a = bq + r$, $r = a - bq$. Suppose x is a factor of a and b , then $\exists y$ and z such that $a = xy$, $b = xz$. Hence: $r = xy - xzq$, $r = x(y - zq)$. x is a factor of r (and also of b and r).

Let $r = a \bmod b$. $\text{hcf}(a, b) = \text{hcf}(b, r)$ because all factors of a and b are also factors of b and r and vice versa. If they have the same factors, they have the same highest common factor.

8

7

What does saying that algorithm A runs in time g mean?

Given an input of size n , the number of operations executed by A is bounded above by $g(n)$.