Where does the word 'Algorithm' come from?

Al-Khwarizmi

c. 750 AD.

Algebra

Kitab Al-Jabr wa'l-Mugabala

Leonardo da Vinci \( \to \) Fibonacci

Fibonacci series

\[ F(n) = 0, 1, 1, 2, 3, 5, 8, 13, \ldots \]

\( n = 0, 1, 2, 3, 4, 5, 6, 7 \)

\( n^{th} \) position/term

how large is \( F(n) \)? Roughly \( 0.7^n \)

\[ F(n) = F(n-1) + F(n-2) \]

\[
\begin{align*}
F(1) &= 1 \\
F(0) &= 0
\end{align*}
\]
\[ \text{fib1}(n) \]
\[ \begin{align*}
\text{if } n &= 0 \quad \text{return } 0 \\
\text{if } n &= 1 \quad \text{return } 1 \\
\text{return } \text{fib1}(n-1) + \text{fib1}(n-2) 
\end{align*} \]

\[ F(n) \]
\[ F(n-1) \]
\[ F(n-2) \]
\[ F(n-3) \]
\[ F(n-4) \]
\[ F(n-5) \]
\[ F(n-6) \]

\[ F(0) = 0(2^n) \]

\[ \sum_{i=0}^{n-1} 2^i = \Theta(2^n) \]

\[ 1 + 2 + 2^2 + \cdots + 2^k = k \]
\[
\frac{1}{\gamma} = \frac{\gamma^{k+1} - 1}{\gamma - 1}, \quad \gamma > 1
\]

\[
1 + \frac{1}{\gamma} + \frac{1}{\gamma^2} + \frac{1}{\gamma^3} + \ldots = \sum_{i=0}^{k} \frac{1}{\gamma^i} = \frac{1}{1-\gamma}, \quad \gamma < 1
\]

\[
\text{fib}_2(n) :
\]
- If \( n = 0 \) or \( n = 1 \) return \( n \)
- \( F(0) = 0 \)
- \( F(1) = 1 \)
- For \( i = 2 \) to \( n \)
  - \( F(i) = F(i-1) + F(i-2) \)
- Return \( F(n) \)

Number of additions: \( O(n) \)

Time complexity versus space complexity

- \# of "basic" operations
- Memory/disk

\[
F(100000) \approx 2^{10^5}
\]

Value of the \( n \)th term

n bits to represent
How long does it take to add two m-bit numbers?

$$\begin{array}{c}
11 \\
101 \\
011 \\
\hline
1000 \\
\hline
\end{array}$$

0(n) time

\[ \frac{m}{64} \]

\[ O(n) \times n = O(n^2) \]

Job 2

\[ O(n) \text{ additions} \]

we are adding 2 n-bit numbers

\[ f(n) \in O(g(n)) \]

big oh \rightarrow asymptotic

\[ n \rightarrow \infty \]

\[ f = 5n^2 \]

\[ g_1 = 2n^2 \]

\[ g_2 = \frac{1}{25} n^3 \]

f and g are two functions
Big Oh def

\[ f(n) = \mathcal{O}(g(n)) \iff \exists \text{ a constant } c \text{ such that for all } n > n_0 \]
\[ f(n) \leq c \cdot g(n) \]

Informally, \[ f(n) \leq g(n) \]

\[ f = 5n^2 + 10n + 5 \]
\[ g_1 = 2n^2 \]

\[ 5n^2 + 10n + 5 \leq 2 \cdot n^2 \]

Free to choose \( n_0 \)

\[ f = \mathcal{O}(g_1) \]
\[ g_1 = \mathcal{O}(f) \] (easy to show)

\[ 2n^2 \leq 1 \left( 5n^2 + 10n + 5 \right) \]
\[ f = O(g) \]

\[ 2n^2 \leq 1(5n^2 + 10n + 5) \]

\[ n_0 = 1 \]

**Complexity Classes**

- \( O(1) \) \(<\) Constant time
- \( O(\log n) \) \(\leq\) Logarithmic (sub-linear time)
- \( O(n) \) \(\leq\) Linear
- \( O(n \log n) \) \(\leq\) Linearithmic
- \( O(n^2) \) \(\leq\) Quadratic
- \( O(n^k) \) \(\leq\) Polynomial
- \( O(2^n) \) \(\leq\) Exponential
- \( O(3^n) \)

\( n! \) \(\leq\) Factorial