Recursion

Christophe Rhodes

Motivation

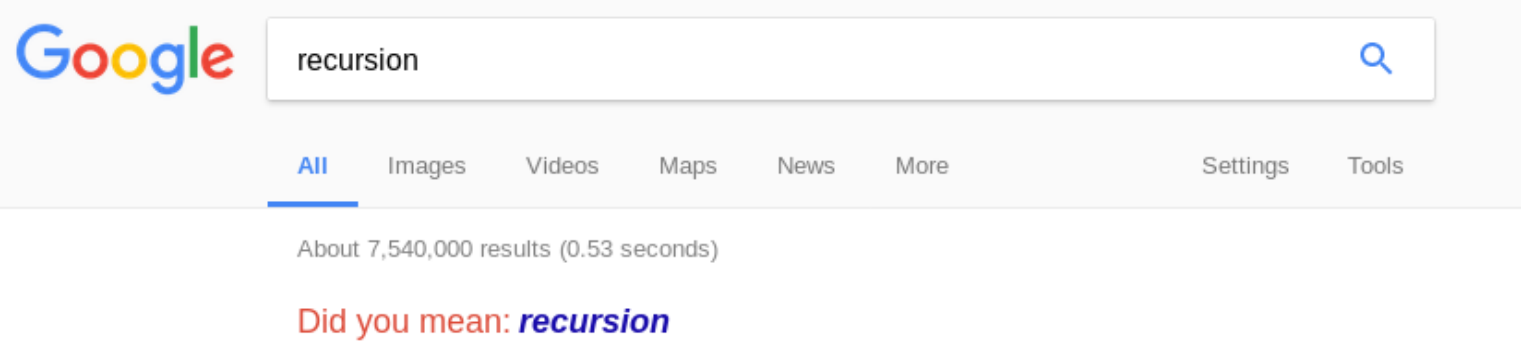
A way to describe solutions of problems that makes them

- easy to prove correct
- easy to compute how they scale

Definition

The definition of a problem or solution in terms of (variant forms of) itself

Illustration



Illustration

– M.C. Escher, *Print Gallery* (1956)

Ingredients

base case non-recursive condition (possibly more than one)

recursive steps rules reducing the problem towards the base case

Examples

factorial $n! = n \times (n-1)!$ and $0! = 1$

fibonacci numbers $F(n) = F(n-1) + F(n-2)$ and $F(0) = 0$, $F(1) = 1$

Tower of Hanoi audience participation!!

Examples: list algorithms

Search

Is the object o present in the list l?

base case is the object o present in the list NIL?

recursive step is the object o equal to the first element of the list? If not, is it in the rest of the list?

Examples: list algorithms

Selection

Return the maximum of the objects in the list l

base case what is the maximum element of the empty list?

alternative base case what is the maximum element of a list with one element?

recursive step how does the first element compare with the maximum of the rest of the list?

Examples: list algorithms

Selection

Return the k^{th} biggest of the objects in the list l

base case what is the k^{th} biggest element of a list with k elements?

recursive step how does the first element compare with the k^{th} biggest element of the rest of the list?

base case, second try what are the k^{th} biggest elements of a list with k elements?

recursive step, second try how does the first element compare with the k^{th} biggest elements of the rest of the list?

Work

1. Reading
    - CLRS, section 2.3