

Running Time and Infeasibility

Is it possible to solve every computable function?

A function can be computable, yet be infeasible to compute in a reasonable amount of time.

B. Is a function feasible to compute?

To decide if a problem is computable in a reasonable amount of time, one must consider all possible algorithms. If one of them can carry out the calculation in a reasonable amount of time (and size), the problem is feasible.

Example: Sorting a set of n elements - test 2 algorithms:

1. Method 1 - The selection sort algorithm:

- find the largest of the n elements.
- find the largest of the remaining $n-1$ elements.
- find the largest of the remaining $n-2$ elements.

...

- the smallest element is remaining.

Work required is $n + n-1 + \dots + 2 + 1 = n(n+1)/2$ units

2. Method 2 - Exhaustive listing and search:

- list all permutations (orderings) of the elements.
- pick the one that is sorted.

Work required is $n(n-1)(n-2) \dots (3)(2)(1) = n!$ units

e.g., for $n=5$:

method 1 requires 15 units of work

method 2 requires 120 units of work!