1. Técnicas de programación

Angel Napa

April 8, 2023

Resumen

Efficiency

Recursive Complete Search

Recursive complete search

Efficiency

Time Complexity

- How much time does the program will take to run for a given input?
- How many operations does de program make to run for a given input?
- **(big O notation): O(...)**

Time Complexity

- $f(n) = \#$ operations in the program

Time Complexity

- $f(n) = \#$ operations in the program
- Common Notation: $f(n) = O(g(n))$

Time Complexity

- $f(n) = \#$ operations in the program
- Common Notation: $f(n) = O(g(n))$
- For n sufficiently large $f(n) \leq C \cdot g(n)$, for some $C > 0$

Time Complexity

- $f(n) = \#$ operations in the program
- Common Notation: $f(n) = O(g(n))$
- For n sufficiently large $f(n) \leq C \cdot g(n)$, for some $C > 0$
- $\alpha < \beta \rightarrow n^{\alpha} << n^{\beta}$

Time Complexity

- $f(n) = \#$ operations in the program
- Common Notation: $f(n) = O(g(n))$
- For n sufficiently large $f(n) \leq C \cdot g(n)$, for some $C > 0$
- $\alpha < \beta \rightarrow n^{\alpha} << n^{\beta}$
- $a > 1 \rightarrow \log_a(n) << n^{\alpha} << a^n$

Time Complexity

$$O(1) << O(\log_2(n)) << O(\sqrt{n}) << O(n) << O(n \cdot \log_2(n))$$

$$<< O(n^2) << O(n^2 \cdot \log_2(n)) << O(n^3) << O(2^n) << O(n!)$$

Time Complexity

Input size	Expected time complexity
$n \leq 10$	$O(n!)$
$n \leq 20$	$O(2^n)$
$n \leq 500$	$O(n^3)$
$n \leq 5000$	$O(n^2)$
$n \leq 10^6$	$O(n \log n)$ or $O(n)$
n is large	$O(1)$ or $O(\log n)$

Recursive Complete Search

Complete Search

- Kattis - Closest Sums
- Each query gives you a number and asks to find the closest sum of 2 elements of a vector.
- $O(n^2 \cdot m)$

Complete Search

- We already saw complete search using iteratives(for, while)
- Sometimes is more efficient solve a problem recursively

Recursive Complete Search

Some examples:

- Generate all subsets of n items
- Generate all permutations of n items
- Generate all ways to put n queens on a $n \times n$ chessboard with no queen atacking eachother.

Bitmasking

- Uva 12455 - Bars
- Given a list l containing $1 \leq n \leq 20$ integers, is there a subset of list l that sums to another given integer X
- We can try all 2^n possible subsets of integers, sum the selected integers for each subset in $O(n)$, and see if the sum of the selected integers equals to X
- Overall time complexity is thus $O(n \cdot 2^n)$
- When $n = 20$, this is just $20 \times 2^{20} \sim 21M$

Recursive complete search

Simple Backtracking

- UVa 750 - 8 Queens Chess Problem
- In chess (with an 8×8 board), it is possible to place eight queens on the board such that no two queens attack each other.
- Determine all such possible arrangements given the position of one of the queen

Gracias