Discrete Mathematics: Algorithm

Nguyễn Gia Phong—BI9-184

December 10, 2019

1 Problem 3

The program `coin.c` takes an integer n from `stdin` and print the least coin exchange of n cents to `stdout`.

2 Problem 4

The program `schedule.c` take an integer n and n integral time intervals from `stdin` and print to `stdout` the chosen talks intervals in chronological order.

3 Problem 5

`search.c` contains two searching implementations, linear search (`lsearch`) and binary search (`bsearch`).

It is trivial that `lsearch` has `nmemb` or $\Theta(n)$ time complexity.

For `binary_search` (which is wrapped by `bsearch`), the time complexity (in term of number of comparisons) is can be seen as

$$\begin{aligned} T(n) &= T\left(\frac{n}{2}\right) + \Theta(1) \\ &= T\left(\frac{n}{2}\right) + \Theta\left(n^{\log_2 1}\right) \end{aligned}$$

since `mid = (lo + high) / 2)`.

By the master theorem[1],

$$T(n) = \Theta\left(n^{\log_2 1} \lg n\right) = \Theta(\lg n)$$

4 Copying

[1]Let $a \geq 1$ and $b > 1$ be constants, and let $T(n)$ be defined on the nonnegative integers by the recurrence

$$T(n) = aT\left(\frac{n}{b}\right) + \Theta\left(n^{\log_b a}\right)$$

where n/b is interpreted as either $\lfloor n/b \rfloor$ or $\lceil n/b \rceil$, then

$$T(n) = \Theta\left(n^{\log_b a} \lg n\right)$$