Australian National University

COMP3600/6466 Algorithms

Lecture 4

S2 2016

Dr. Hassan Hijazi

Prof. Weifa Liang

Algorithm Analysis

Problem

New Algorithm

Existing Algorithms

Australian National University

Algorithm Analysis

$$\xrightarrow{\quad ? \quad} \quad \Theta(n^3 \ln n)$$

Pseudocode Analysis

Algorithm 1 My algorithm

```
procedure MyProcedure
    stringlen ← length of string
    i ← patlen
top:
    if i > stringlen then return false
    j ← patlen
loop:
    if string(i) = path(j) then
        j ← j − 1.
        i ← i − 1.
        goto loop.
        close;
    i ← i + max(delta_1(string(i)), delta_2(j)).
    goto top.
```

Pseudocode Analysis

Iterative

Algorithm 1 My algorithm

procedure MyProcedure
    stringlen $\leftarrow$ length of *string*
    $i \leftarrow patlen$
top:
    if $i > stringlen$ **then return** false
    $j \leftarrow patlen$
loop:
    if $string(i) = path(j)$ **then**
        $j \leftarrow j - 1.$
        $i \leftarrow i - 1.$
        goto *loop*.
        close;
    $i \leftarrow i + \max(delta_1(string(i)), delta_2(j)).$
    goto *top*.

Recursive

Algorithm 1 My algorithm

procedure MyProcedure
    stringlen $\leftarrow$ length of *string*
    $i \leftarrow patlen$
top:
    if $i > stringlen$ **then return** false
    $j \leftarrow patlen$
loop:
    if $string(i) = path(j)$ **then**
        $j \leftarrow j - 1.$
        $i \leftarrow i - 1.$
        goto *loop*.
        close;
    $i \leftarrow i + \max(delta_1(string(i)), delta_2(j)).$
    goto *top*.

Iterative Algorithms and Summations

Total running time
=
Sum of times in each execution of the body of the loop.

Given a sequence $t_1, t_2, \ldots, t_n$,

the finite sum $t_1 + t_2 + \cdots + t_n$ is written as $\displaystyle\sum_{k=1}^{n} t_k$.

Common Sums in Algorithm Analysis

1. Geometric sums: $\displaystyle\sum_{k=1}^{n} r^k$, decreasing if $0 < r < 1$, increasing if $r > 1$.

2. Power sums: $\displaystyle\sum_{k=1}^{n} k^t$ for any constant $t > 0$.

3. Harmonic sum: $\displaystyle\sum_{k=1}^{n} \frac{1}{k}$.

Geometric Sums

$$S_n = \sum_{k=0}^{n-1} a_1 r^k,\ a_1 > 0$$

Given a geometric sequence $a_1, a_2, \ldots, a_n$ with $a_1 > 0$, let $r = \frac{a_{i+1}}{a_i}$, then

$$a_{i+1} = ra_i = r^2 a_{i-1} = r^3 a_{i-2} = \ldots = r^i a_1$$

Let S_n be the sum of all elements in this sequence:

$$\begin{aligned} S_n &= a_1 + a_2 + a_3 + \ldots + a_{n-1} + a_n, \\ &= a_1(1 + r + r^2 + \ldots + r^{n-1}). \end{aligned}$$

$$\begin{aligned} rS_n &= r(a_1 + a_2 + a_3 + \ldots + a_{n-1} + a_n), \\ &= a_1(r + r^2 + r^3 + \ldots + r^{n-1} + r^n), \end{aligned}$$

Geometric Sums

We have

$rS_n - S_n = [a_1(r + r^2 + r^3 + \ldots + r^{n-1}) + a_1 r^n] - a_1(1 + r + r^2 + \ldots + r^{n-1}),$

$(r-1)S_n = a_1 r^n - a_1 = a_1(r^n - 1),$

$$S_n = \frac{a_1(r^n - 1)}{r - 1}$$

Let's prove that $S_n = \Theta(\text{largest term})$

Complete proof by hand

Geometric Sums

$$S_n = \frac{a_1(r^n - 1)}{r - 1}$$

Let's prove that $S_n = \Theta(\text{largest term})$

$$\Updownarrow$$

$S_n = \Theta(1)$ if $0 < r < 1$ and $S_n = \Theta(r^n)$ if $r \geq 1$

Exercise 4.1

Give a tight asymptotic bound for:

$$f(n) = \sum_{k=1}^{n} 4^k$$

Complete proof by hand

What if r depends on k?

$$\text{Let } f(n) = \sum_{k=1}^{n} \frac{k^2}{3^k}$$

We can still bound the sum using a geometric sequence!

$$r = \frac{a_{k+1}}{a_k} = \frac{3^k\,(k+1)^2}{3^{k+1}\,k^2} = \frac{1}{3}\left(1 + \frac{1}{k}\right)^2$$

$$\text{Is } 0 < r < 1 \text{ or } r \geq 1?$$

$$\text{We have } r \leq \frac{3}{4} < 1,\ \forall k \geq 2$$

What if r depends on k?

$$\text{We have } r = \frac{a_{k+1}}{a_k} \leq \frac{3}{4}, \ \forall k \geq 2$$

$$\Downarrow$$

$$a_{k+1} \leq a_k \frac{3}{4} \leq a_{k-1} \left(\frac{3}{4}\right)^2 \leq \ldots \leq a_2 \left(\frac{3}{4}\right)^{k-1}, \ \forall k \geq 2$$

$$\Downarrow$$

$$a_k \leq a_2 \left(\frac{3}{4}\right)^{k-2}, \ \forall k \geq 2$$

Complete the proof by hand

Power Sums

$$S_n = \sum_{k=1}^{n} k^t$$

$$n \times \text{smallest term} \leq S_n \leq n \times \text{largest term}$$

$$n \times 1 \leq S_n \leq n \cdot n^t$$

$$n \leq S_n \leq n^{t+1}$$

$$S_n = \Omega(n) \text{ and } S_n = O(n^{t+1})$$

Power Sums

$$S_n = \Omega(n) \text{ and } S_n = O(n^{t+1})$$

$$\text{Is } S_n = \omega(n)?$$

YES!

Can we find a tight asymptotic bound?

$$n \times \text{smallest term} \leq S_n \leq n \times \text{largest term}$$

$$\text{biggest half of the sum} \leq S_n \leq n \times \text{largest term}$$

$$\frac{n}{2} \times \text{smallest element in the biggest half of the sum} \leq S_n \leq n \times \text{largest term}$$

Power Sums

$$S_n = \Omega(n) \text{ and } S_n = O(n^{t+1})$$

Can we find a tight asymptotic bound?

$$\text{biggest half of the sum} \leq S_n \leq n \times \text{largest term}$$

$$\frac{n}{2} \times \text{smallest element in the biggest half of the sum} \leq S_n \leq n \times \text{largest term}$$

Complete the proof by hand

Summation Splitting Technique

Exercise 4.2

Using summation splitting, give a tight asymptotic bound for:

$$f(n) = \sum_{k=1}^{n} k^2 \ln k$$

Mathematical Induction

To prove that S_n has a property P:

Base case: Prove that there is a constant n for which P is true.

Induction: Assume that P holds for n, then show that it also holds for $n+1$.

Exercise 4.3

Show that $\sum_{k=1}^{n} k^5 = \Omega(n^6)$ using mathematical induction

Complete the proof by hand