ECS 20 — Lecture 15 — Fall 2013 —14 Nov 2013

Phil Rogaway

Today:

- Solving recurrence relations .
- Pigeonhole arguments

Announcements:

- Quiz 3 on Tuesday

Karatsuba algorithm (1960/1962) Suppose we want to multiply two decimal numbers. We write one number as $x = x_1 \mathbin{||} x_0$ and the other was $y = y_1 \mathbin{||} y_0$, each half having m digits (let's not worry about what to do if m is odd; no real complications are added). So

$x = x_1\, 10^m + x_0$
$y = y_1\, 10^m + y_0$,

The product is then
$xy = (x_1 10^m + x_0)(y_1 10^m + y_0)$
$\quad = z_2 10^{2m} + z_1 10^m + z_0$
where
$z_2 = x_1 y_1$
$z_1 = x_1 y_0 + x_0 y_1$
$z_0 = x_0 y_0$.

These formulas require **four multiplications**. Karatsuba observed that xy can be computed in only **three multiplications** of m-digit values. With z_0 and z_2 as before we can calculate
$z_1 = (x_1 + x_0)(y_1 + y_0) - z_2 - z_0$
which holds since
$z_1 = (x_1 + x_0)(y_1 + y_0) - x_1 y_1 - x_0 y_0 = x_1 y_0 + x_0 y_1$

Example Let's compute

```
    98  76
 ×  56  78
---------
      5928
    7644
    4256    these two sum to 11900.   But we can also get 11900 as
  5488      11900 = (98+76)(56+78) - 5928 - 5488
---------         =        174*134 - 5928 - 5488
 56075928         =          23316 - 5928 - 5488
                  = 11900
```

Comparing the asymptotic running times

First, the 4-multiply method:

$T(n) = 4T(n/2) + n$ (when $n > 1$; $T(n) = \text{const}$ when $n = 1$)

$= 4(4T(n/4) + n/2) + n$

$= 4^2\, T(n/4) + 2n + n$

$= 4^3\, T(n/8) + n(1 + 2 + 4)$

$= 4^4\, T(n/2^4) + n(1 + 2 + 2^2 + 2^3)$

$= \ldots$

$= 4^k + n(2^k - 1)$

$\in \Theta(n) + O(n^2)$

$\in \Theta(n^2)$

Now, the 3-multiply method:

$T(n) = 3\, T(n/2) + n$

$= 3\,(3T(n/4) + n/2) + n$

$= 3^2\, T(n/4) + (3n/2 + n)$

$= 3^2(3T(n/8) + n/4) + 3n/2 + n$

$= 3^3\, T(n/8) + 3^2 n/2^2 + 3n/2 + n$

$= 3^3\, T(n/8) + n(1 + 3/2 + (3/2)^2))$

$= 3^4\, T(n/16) + n\,(1 + (3/2) + (3/2)^2 + (3/2)^3)$

$= \ldots$

$= 3^k\, T(n/2^k) + n\,(1 + (3/2) + (3/2)^2 + (3/2)^3 + \ldots + (3/2)^{k-1})$

At this point it would be good to know what is

$S = 1 + x + x^2 + \ldots + x^{k-1} + x^k$

$Sx = x + x^2 + \ldots + x^k + x^{k+1}$

$1+Sx = 1 + x + x^2 + \ldots + x^k + x^{k+1}$

$1+Sx = S + x^{k+1}$

$S(x-1) = x^{k+1} - 1$

$S = (x^{k+1} - 1) \;/\; (x-1)$

It is worth remembering this result (or, better, being able to re-derive it if you need it).

$1 + x + x^2 + \ldots + x^{k-1} = (x^k - 1) \;/\; (x-1)$

So, with $x = 3/2$, we have

$(1 + (3/2) + (3/2)^2 + (3/2)^3 + \ldots + (3/2)^{k-1}) = 2\,(3/2)^k - 2$

$= 3^k\, T(n/2^k) + n\,(1 + (3/2) + (3/2)^2 + (3/2)^3 + \ldots + (3/2)^{k-1})$

$= 3^k\, T(n/2^k) + n\;((3/2)^k - 1) / (1/2)$

Now we want $k = \lg n$, so

$= 3^{\lg n} + 2n\,(3/2)^{\lg n}$

$= (2^{\lg 3})^{\lg n} + 2n\; 3^{\lg n} / 2^{\lg n}$

$= n^{\lg 3} + 2\; n^{\lg 3}$

$= \Theta(n^{\lg 3})$

$= \Theta(n^{1.5849})$

Best-known: we can actually multiply two n-digit numbers in time $\Theta(n \log n \log\log n)$)or this number of 2-input gaates) using the Schönhage–Strassen algorithm (1971) – the third multiplicand not improved by Fürer's (2007)

Notation	Intuition	Informal definition: for sufficiently large n...	Formal Definition
$f(n) \in O(g(n))$	f is bounded above by g (up to constant factor)	$\|f(n)\| \leq g(n) \cdot k$ for some positive *k*	$\exists k > 0 \; \exists n_0 \; \forall n > n_0 \; f(n) \leq g(n) \cdot k$
$f(n) \in \Omega(g(n))$	f is bounded below by g	$f(n) \geq g(n) \cdot k$ for some positive *k*	$\exists k > 0 \; \exists n_0 \; \forall n > n_0 \; g(n) \cdot k \leq f(n)$
$f(n) \in \Theta(g(n))$	f is bounded above and below by g	$g(n) \cdot k_1 \leq f(n) \leq$ for some positive k_1, k_2	$\exists k_1 > 0 \; \exists k_2 > 0 \; \exists n_0 \; \forall n > n_0$ $g(n) \cdot k_1 \leq f(n) \leq g(n) \cdot k_2$
$f(n) \in o(g(n))$	f is dominated by g	$\|f(n)\| \leq k \cdot \|g(n)\|$ for every fixed positive number k	$\forall k > 0 \; \exists n_0 \; \forall n > n_0 \; \|f(n)\| \leq k \cdot \|g(n)\|$
$f(n) \in \omega(g(n))$	f dominates g	$\|f(n)\| \geq k \cdot \|g(n)\|$ for every fixed positive number k	$\forall k > 0 \; \exists n_0 \; \forall n > n_0 \; \|f(n)\| \geq k \cdot \|g(n)\|$
$f(n) \sim g(n)$	f is equal to g asymptotically	$f(n)/g(n) \to 1$	$\forall \varepsilon > 0 \; \exists n_0 \; \forall n > n_0 \; \left\| \frac{f(n)}{g(n)} - 1 \right\| < \varepsilon$

If an additional example feels needed: do **mergesort**

The asymptotic "debate"

Asymptotic notation is everywhere in computer science, but not everyone is a fan.

Reasons for asymptotic notation:

1. **Simplicity** – makes arithmetic simple, makes analyses easier
2. Applied to running times: Works well, in practice, to get an **understanding of efficiency**
3. When applied to running times: Facilitates greater **model-independence**

Reasons against:

1. Hidden constants **can** matter
2. Excessive reliance on asymptotics: may fail to notice about things that one really **should** care about
3. Not everything has an "*n*" value to grow with respect to – or, may really be interested in one particular *n*.

There is more than O and Θ. (Table modified from Wikipedia)

Back to the Pigeonhole Principle

If N pigeons roost in n holes, N>n, then some two pigeons share a hole.

Restated: [Pigeonhole principle]
If $f: A \to B$ where A and B are finite sets, $|A|>|B|$, then f is NOT injective.

Or
[Pigeonhole principle, strong form]
If $f: A \to B$ where A and B are finite sets, then so point $b \in B$ must have at least $\lceil |B|/|A| \rceil$ preimages.

Eg, if 100 pigeons roost in 30 holes, some hole has at least 4 pigeons roosting therein.

Ex 0. Any room with 3 or more people has some two of the same gender.

Ex 1. 20 people at a party, some two have the same number of friends.
number of friends
proof: 0..18 or 1..19

Ex 2: Given five points inside the square whose side is of length 2, prove that two are within \sqrt{2} of each other.

Soln: divide square into four 1 x 1 cells. Diameter of each cell = \sqrt{2}

Ex 3: In any list of **10** numbers, a_1, ..., a_10, there's a subsequence of (consecutive) numbers whose sum is divisible by 10.

Consider

s_1 = a_1
s_2 = a_1 + a_2
...
s_10 = a_1 + a_2 + ... + a_10

Then numbers in the list. If any of these divisible by 10: done.

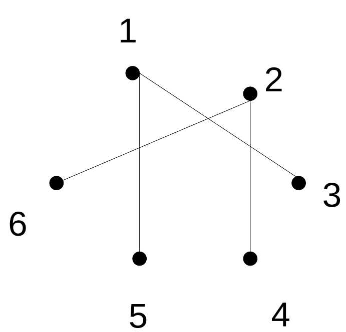
Otherwise, each is **congruent** to 1,..., 9 mod 10. So two of the s_i (mod 10) values are congruent to the **same** thing. Eg, may

a_1 + a_2 + a_3 = 6 (mod 5)
a_1 + a_2 + a_3 + a_4 + a_5 = 6 (mod 5)

But then

a+4 + a_5 = 0 (mod 10)

Ex 4. (beautiful example) In any room of 6 people, there are 3 mutual friends or 3 mutual strangers (Ramsey theorem, and R(3,3)=6)



Remove person 1 5 people left.
Put into two pots: friends with 1, non-friends with 1.
One has at least three people.
If three friends: Case 1: some two know each other: DONE
Case 2: no two know each other: DONE
If three non-friends: ...o

Difficult Puzzle: What is the minimum number of people that must assemble in a room such that there will be at least n friends or n non-friends: R(n,n)

R(4,4) = 18 (1955)
R(5,5) = ?? open!!! known to be between 43 (1989) and 49 (1995)
R(10,10) =?? open and not tightly determined at all: range 798 (1986)- 23,556 (2002)