COS 423: Theory of Algorithms

Princeton University • COS 423 • Theory of Algorithms • Spring 2001 • Kevin Wayne

Last Lecture

Perspective and course review.

Top 10 scientific algorithms.

Course evaluations.

Final exercises due Tuesday, May 15 at 5pm.

- Individual write-ups.
- Collaboration allowed.

Theory of Algorithms

Algorithm. (webster.com)

- A procedure for solving a mathematical problem (as of finding the greatest common divisor) in a finite number of steps that frequently involves repetition of an operation.
- Broadly: a step-by-step procedure for solving a problem or accomplishing some end especially by a computer.

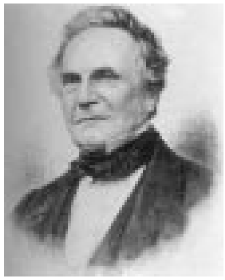
Etymology.

- "algos" = Greek word for pain.
- "algor" = Latin word for to be cold.
- Abu Ja'far al-Khwarizmi's = 9th century Arab scholar.
  - his book "Al-Jabr wa-al-Muqabilah" evolved into today's high school algebra text

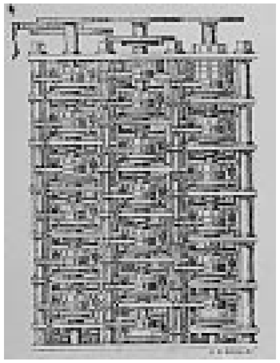
Theory of Algorithms

A strikingly modern thought.

"As soon as an Analytic Engine exists, it will necessarily guide the future course of the science. Whenever any result is sought by its aid, the question will arise - By what course of calculation can these results be arrived at by the machine in the shortest time?"



Charles Babbage (1864)



Why Does It Matter?

Run time (nanoseconds)		$1.3\ N^3$	$10\ N^2$	$47\ N \log_2 N$	$48\ N$
Time to solve a problem of size	1000	1.3 seconds	10 msec	0.4 msec	0.048 msec
	10,000	22 minutes	1 second	6 msec	0.48 msec
	100,000	15 days	1.7 minutes	78 msec	4.8 msec
	million	41 years	2.8 hours	0.94 seconds	48 msec
	10 million	41 millennia	1.7 weeks	11 seconds	0.48 seconds
Max size problem solved in one	second	920	10,000	1 million	21 million
	minute	3,600	77,000	49 million	1.3 billion
	hour	14,000	600,000	2.4 trillion	76 trillion
	day	41,000	2.9 million	50 trillion	1,800 trillion
N multiplied by 10, time multiplied by		1,000	100	10+	10

Orders of Magnitude

Seconds	Equivalent
1	1 second
10	10 seconds
10^2	1.7 minutes
10^3	17 minutes
10^4	2.8 hours
10^5	1.1 days
10^6	1.6 weeks
10^7	3.8 months
10^8	3.1 years
10^9	3.1 decades
10^{10}	3.1 centuries
. . .	forever
10^{21}	age of universe

Meters Per Second	Imperial Units	Example
10^{-10}	1.2 in / decade	Continental drift
10^{-8}	1 ft / year	Hair growing
10^{-6}	3.4 in / day	Glacier
10^{-4}	1.2 ft / hour	Gastro-intestinal tract
10^{-2}	2 ft / minute	Ant
1	2.2 mi / hour	Human walk
10^2	220 mi / hour	Propeller airplane
10^4	370 mi / min	Space shuttle
10^6	620 mi / sec	Earth in galactic orbit
10^8	62,000 mi / sec	1/3 speed of light

Powers of 2	2^{10}	thousand
	2^{20}	million
	2^{30}	billion

What was COS 423?

Introduction to design and analysis of computer algorithms.

- Algorithmic paradigms.
- Analyze running time of programs.
- Understand fundamental algorithmic problems.
- Intrinsic computational limitations.
- Models of computation.
- Critical thinking.

Material Covered

Algorithmic paradigms.

- Divide-and-conquer.
- Greed.
- Dynamic programming.
- Reduction.

Analysis of algorithms.

- Recurrences and big Oh.
- Amortized analysis.
- Average-case analysis.

Other models of computation.

- On-line algorithms.
- Randomized algorithms.

Intractability.

- Polynomial reductions.
- NP completeness.
- Approximation algorithms.

Fundamental algorithmic problems.

- Sorting and searching.
- Integer arithmetic.
- FFT.
- MST.
- Shortest path.
- Max flow.
- Linear programming.

Brief History of Algorithms

300 B. C.
- Euclid’s gcd algorithm.

780-850 A.D.
- Abu Ja’far Mohammed Ben Musa al-Khwarizmi.

1424 A.D.
- π = 3.1415926535897932…

1845.
- Lamé: Euclid's algorithm takes at most $1 + \log_\phi (n \sqrt{5})$ steps.

1900.
- Hilbert's 10th problem.

1910.
- Pocklington: bit complexity.

1920-1936.
- Post, Goëdel, Church, Turing.

1965.
- Edmonds: polynomial vs. exponential algorithms.

1971.
- Cook's Theorem, Karp reductions.

20xx.
- $P \neq NP$???

Top 10 Scientific Algorithms of 20th Century

Computing in Science and Engineering. (January, 2000).

"the greatest influence on the development and practice of science and engineering in the 20th century"

"For me, great algorithms are the poetry of computation. Just like verse, they can be terse, allusive, dense, and even mysterious. But once unlocked, they cast a brilliant new light on some aspect of computing."

-Francis Sullivan

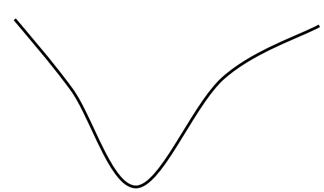
Top 10 Scientific Algorithms of 20th Century

1. Metropolis Algorithm/ Monte Carlo method (von Neumann, Ulam, Metropolis, 1946). Through the use of random processes, this algorithm offers an efficient way to stumble toward answers to problems that are too complicated to solve exactly.
   - Approximate solutions to numerical problems with too many degrees of freedom.
   - Approximate solutions to combinatorial optimization problems.
   - Generation of random numbers.

Metropolis Algorithm

Local search. Algorithm that explores the space of possible solutions in sequential fashion, moving in one step from a current solution to a "nearby" one.
- TSP: given a tour, perturb it by exchanging order of two cities.
- VERTEX-COVER: given a vertex cover, perturb it by adding or deleting a node, so that resulting set remains a cover.

Gradient descent. Replace current solution with neighboring solution that improves objective function, until no such neighbor exists.



A funnel

A jagged funnel

Metropolis Algorithm

Metropolis algorithm. Gradient descent, but occasionally replace current solution with "uphill" solution.

- Simulate behavior of system according to principles of statistical mechanics.
- Probability of finding a physical system in a state with energy E is proportional to Gibbs-Boltzmann function $e^{-E/(kT)}$, where T > 0 is temperature and k is a constant.

Theorem. Let $f_S(t)$ be fraction of first t steps in which state of simulation in in state S $\in \Sigma$. Then, with probability 1:

$$\lim_{t \to \infty} f_S(t) = \tfrac{1}{Z} e^{-E(S)/(kT)},$$

$$\text{where } Z = \sum_{S \in \Sigma} e^{-E(S)/(kT)}.$$

Metropolis Step(S)

```
Find neighboring solution S'.
IF (c(S') ≤ c(S))
   Update S ← S'.
ELSE
   E ← c(S') - c(S).
   Update S ← S' with
      probability e - E /(kT).
```


Metropolis Algorithm

Simulated annealing.

- T large $\Rightarrow$ probability of accepting an uphill move is large.
- T small $\Rightarrow$ uphill moves are almost never accepted.
- Idea: turn knob to control T.
- Cooling schedule: T = T(i) at iteration i.

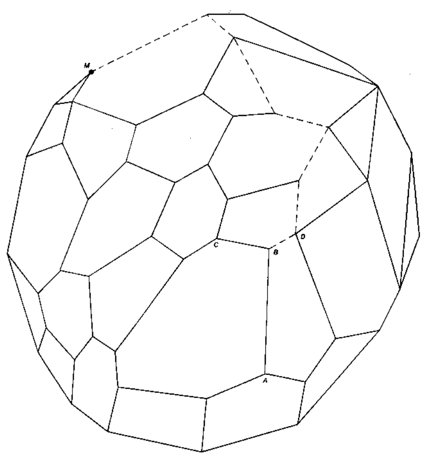
Physical analog.

- Take solid and raise it to high temperature, we do not expect it to maintain a nice crystal structure.
- Take a molten solid and freeze it very abruptly, we do not expect to get a perfect crystal either.
- Annealing: cool material gradually from high temperature, allowing it to reach equilibrium at succession of intermediate lower temperatures.

Top 10 Scientific Algorithms of 20th Century

2. Simplex Method for Linear Programming (Dantzig 1947).
An elegant solution to a common problem in planning and decision-making: max {cx : Ax $\leq$ b, x $\geq$ 0}.

- One of most successful algorithms of all time.
- Dominates world of industry.



Top 10 Scientific Algorithms of 20th Century

3. Krylov Subspace Iteration Method (Hestenes, Stiefel, Lanczos, 1950).
A technique for rapidly solving Ax = b where A is a huge n x n matrix.

- Conjugate gradient method for symmetric positive definite systems.
- GMRES, CGSTAB for non-symmetric systems.

Compute $r^{(0)} = b - Ax^{(0)}$ for some initial guess $x^{(0)}$
for $i = 1, 2, \ldots$
  solve $Mz^{(i-1)} = r^{(i-1)}$
  $\rho_{i-1} = r^{(i-1)^T} z^{(i-1)}$
  if $i = 1$
    $p^{(1)} = z^{(0)}$
  else
    $\beta_{i-1} = \rho_{i-1}/\rho_{i-2}$
    $p^{(i)} = z^{(i-1)} + \beta_{i-1} p^{(i-1)}$
  endif
  $q^{(i)} = Ap^{(i)}$
  $\alpha_i = \rho_{i-1}/p^{(i)^T} q^{(i)}$
  $x^{(i)} = x^{(i-1)} + \alpha_i p^{(i)}$
  $r^{(i)} = r^{(i-1)} - \alpha_i q^{(i)}$
  check convergence; continue if necessary
end

Preconditioned Conjugate Gradient

Top 10 Scientific Algorithms of 20th Century

4. Decompositional Approach to Matrix Computations (Householder, 1951). A suite of technique for numerical linear algebra that led to efficient matrix packages.

- Factor matrices into triangular, diagonal, orthogonal, tri-diagonal, and other forms.
- Analysis of rounding errors.
- Applications to least squares, eigenvalues, solving systems of linear equations.
- LINPACK, EISPACK.

Top 10 Scientific Algorithms of 20th Century

5. Fortran Optimizing Compiler (Backus, 1957). Turns high-level code into efficient computer-readable code.

- Among single most important events in history of computing: scientists could program computer without learning assembly.

Fortran Code

```
500   C = 0.0
C   *** START LOOP ***
      DO 540 I=L,K
         F = S*RV1(I)
         RV1(I) = C*RV1(I)
         IF (ABS(F).LE.EPS) GO TO 550
         G = W(I)
         H = SQRT(F*F+G*G)
         W(I) = H
         C = G/H
         S = -F/H
510   CONTINUE
```

Top 10 Scientific Algorithms of 20th Century

6. QR Algorithm for Computing Eigenvalues (Francis 1959). Another crucial matrix operation made swift and practical.

- Eigenvalues are arguably most important numbers associated with matrices.

$$\boxed{\mathbf{A\,x} = \lambda\,\mathbf{x}}$$

- Differential equations, population growth, building bridges, quantum mechanics, Markov chains, web search, graph theory.

QR(A)

```
Initialize A0 = A

FOR k = 0, 1, 2, ...
   Factor   Ak   = Qk Rk
   Compute  Ak+1 = Rk Qk
```

$$\begin{aligned} A_{k+1} &= R_k\,Q_k \\ &= Q_k^{-1}\,Q_k\,R_k\,Q_k \\ &= Q_k^{-1}\,A_k\,Q_k \end{aligned}$$

$\Rightarrow A_{k+1}$ and A_k have same eigenvalues

Under fairly general conditions, A_k converges to diagonal or upper triangular matrix with eigenvalues on main diagonal.

Web Search

AltaVista text-based search for `'+censorship +net'` might yield tens of thousands of hits, ordered as follows:

- *www.epic.org/free_speech/action*
- *www.zepa.net/hypermail/asfar/1998/07/0466.html*
- *www.eserver.org/internet/censorship.html*
- *www.tiac.net/users/sojourn/censor0596.html*
- *www.anatomy.usyd.edu.au/danny/usenet/aus.net.news/*

Abundance problem: number of pages that can be returned as relevant is far too large for human to digest.

- Observation: not many useful pages here.

Web Search

Some "authoritative" pages (obtained from Kleinberg algorithm):

- www.eff.org (Electronic Frontier Foundation)
- www.cdt.org (Center for Democracy and Technology)
- www.vtw.org (Voters Telecommunications Watch)
- www.aclu.org (American Civil Liberties Union)

Authoritative page: need quantitative definition.

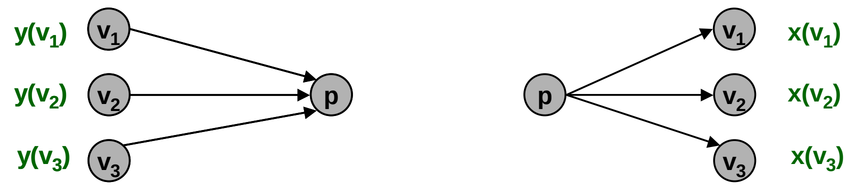
- Non-trivial problem: query for "search engine" unlikely to report Yahoo, Excite, or AltaVista since they do not use the term.
- Yahoo solution: legion of human catalogers.
- Elegant solution (Kleinberg, Google): use latent human judgment implicit in hyperlink structure of Web.
  - page p points to q: creator of page p confers authority on q
  - pitfalls: navigational links, relevance vs. popularity

Hubs and Authorities

Good hub: page that points to many good authorities.
Good authority: page pointed to by many good hubs.

Iterative algorithm: authority weights x(p), and hub weights y(p).

- Set authority weights x(p) = 1, and hub weights y(p) = 1 for all p.
- Repeat following two operations (and then re-normalize x and y to have unit norm):



$$x(p) = \sum_{q \text{ points to } p} y(q) \qquad y(p) = \sum_{p \text{ points to } q} x(q)$$

Hubs and Authorities

Theorem (Kleinberg, 1997). The iterates x(p) and y(p) converge to the principal eigenvectors of A^TA and AA^T, where A is the adjacency matrix of the (directed) Web subgraph.

- Algorithm is essentially "Power method" for computing principal eigenvector.
- Can use any eigenvector algorithm, e.g., QR algorithm.

Web Search: Clustering

Principal eigenvector.

- www2.ecst.csuhcico.edu/.../jaguar.html (404 Not Found)
- www.mcc.ac.uk/dlms/.../du/.../jaguar.html (Jaguar Page)

2nd non-principal eigenvector: positive components.

- www.jaguarsnfl.com (Jacksonville Jaguars NFL)
- www.nando.net/.../jax.htm (Jacksonville Jaguars Home Page)

3rd non-principal eigenvector: positive components.

- www.jaguarvehicles.com (Jaguar Cars Global Home Page)
- www.collection.co.uk (The Jaguar Collection)

Web Search: Clustering

2^{nd} non-principal eigenvector: positive components.

- *www.caral.org/abortion.html* *(Abortion and Reproductive Rights)*
- *www.plannedparenthood.org* *(Welcome to Planned Parenthood)*
- *www.gynpages.com* *(Abortion Clinics Online)*
- *www.prochoice.org/naf* *(National Abortion Federation)*

2^{nd} non-principal eigenvector: negative components.

- *www.awinc.com/…/lifenet.htm* *(LifeWEB)*
- *www.worldvillage.com/…/peter.htm* *(Healing After Abortion)*
- *www.members.aol.com/pladvocate* *(Pro-Life Advocate)*
- *www.catholic.net/…/abortion.html*

Top 10 Scientific Algorithms of 20th Century

7. Quicksort (Hoare, 1962). Given N items over a totally order universe, rearrange them in increasing order.

- O(N log N) instead of O(N^2).
- Efficient handling of large databases.

8. Fast Fourier Transform (Cooley, Tukey 1965). Perhaps the most ubiquitous algorithm in use today, it breaks down waveforms (like sound) into periodic components.

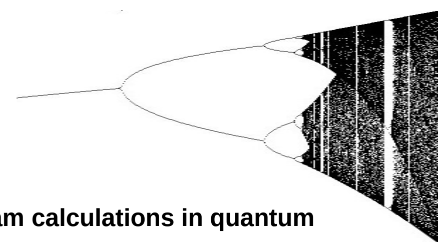
- O(N log N) instead of O(N^2).

Top 10 Scientific Algorithms of 20th Century

9. Integer Relation Detection (Ferguson, Forcade, 1977). Given real numbers $x_1, \ldots, x_n$, find integers $a_1, \ldots, a_n$ (not all 0 if they exist) such that $a_1x_1 + \ldots + a_nx_n = 0$?

- PSLQ algorithm generalizes Euclid's algorithm: special case when n = 2.
- Find coefficients of polynomial satisfied by 3^{rd} and 4^{th} bifurcation points of logistic map.

$$x_{n+1} = a\, x_n\, (1 - x_n)$$

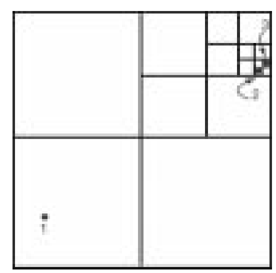


- Simplify Feynman diagram calculations in quantum field theory.
- Compute n^{th} bit of π without computing previous bits.
- Experimental mathematics.

Top 10 Scientific Algorithms of 20th Century

10. Fast Multipole Method (Greengard, Rokhlin, 1987). Accurate calculations of the motions of N particles interacting via gravitational or electrostatic forces.

- Central problem in computational physics.
- O(N) instead of O(N^2).
- Celestial mechanics, protein folding, etc.

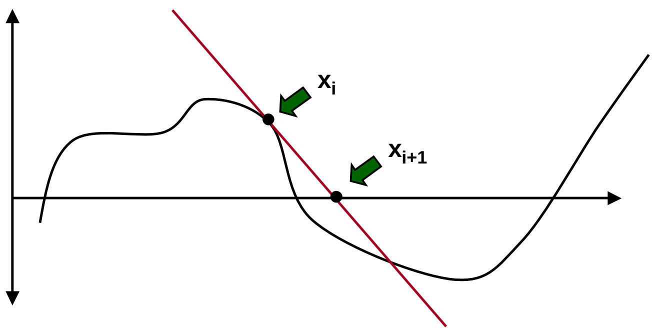


A Quad-Tree

Kevin's Lifetime Achievement Award

11. Newton's method (Newton, 16xx). Given a differentiable function f(x), find a value x* such that f(x*) = 0.

- Start with initial guess x_0.
- Compute a sequence of approximations: $x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$.
- Equivalent to finding line of tangent to curve y = f(x) at x_i and taking x_{i+1} to be point where line crosses x-axis.



Kevin's Lifetime Achievement Award

11. Newton's method (Newton, 16xx). Given a differentiable function f(x), find a value x* such that f(x*) = 0.

- Tabulating square roots, etc.
- Solving systems of nonlinear equations: $x_{i+1} = x_i - J^{-1}(x_i)\, f(x_i)$.
- Continuous optimization: $x_{i+1} = x_i - H^{-1}(x_i)\, \nabla f(x_i)$.
- Integer division.
- Interior point algorithms.

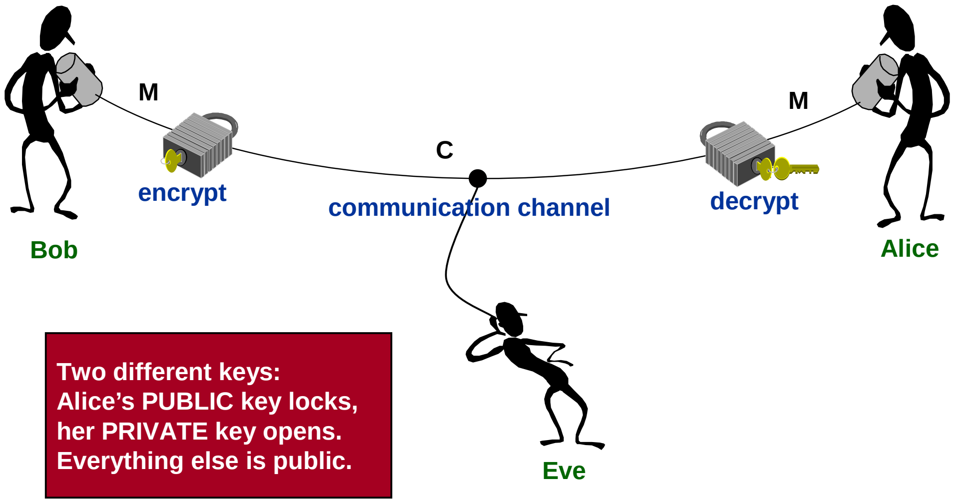
Kevin's Non-Scientific Honorable Mention

12. Depth first search (Tarjan). Learn properties of a graph by systematically examing each of its vertices and edges.

- Connectivity.
- Cycle detection.
- Bipartiteness.
- 2-SAT, 2-colorability.
- Topological sort.
- Transitive closure.
- Euler tour.
- Bi-connectivity.
- Strong connectivity.
- Planarity.

Kevin's Non-Scientific Honorable Mention

13. RSA public-key cryptosystem (Rivest-Shamir-Adleman, 1978). Most widely used public-key cryptosystem: Sun, Microsoft, Apple, browsers, cell phones, ATM machines, . . .



Two different keys:
Alice's PUBLIC key locks,
her PRIVATE key opens.
Everything else is public.

RSA Public-Key Cryptosystem

Key generation.

- Select two large prime numbers p and q at random.
- Compute $n = pq$, and $\phi = (p-1)(q-1)$.
- Choose integer e that is relatively prime to ϕ.
- Compute d such that $d\,e \equiv e\,d \equiv 1 \pmod{\phi}$.
- Publish (e, n) as public key.
- Keep (d, n) as secret key.

p = 11, q = 29
n = 319, ϕ = 280
e = 3, d = 187
M = 100

RSA Public-Key Cryptosystem

Bob sends message M to Alice. $\boxed{M < n}$

- Bob obtains Alice's public key (e, n) from Internet.
- Bob computes $C = M^e \pmod{n}$.

Alice receives message C.

- Alice uses her secret key (d, n).
- Alice computes $M' = C^d \pmod{n}$.

Why does it work? Need M = M'. Intuitively.

- $$\begin{aligned} M' &\equiv C^d && \pmod{n} \\ &\equiv M^{ed} && \pmod{n} \\ &\equiv M && \text{Recall: } e\,d \equiv 1 \pmod{\phi}. \end{aligned}$$

- Argument not rigorous because of mod.
  - rigorous argument uses fact that p and q are prime and $\phi = (p-1)(q-1)$

RSA Example

Parameters.

- $p = 47$, $q = 79$, $n = 3713$, $\phi = 3588$
  $e = 17$, $d = 3377$
- $M = 2003$

2003^{17}	(mod 3713)
= $2003^{16} * 2003^{1}$	(mod 3713)
= 3157 * 2003	(mod 3713)
= 6323471	(mod 3713)
= 232	

Modular exponentiation.

- $2003^{17} \pmod{3713}$
  = 134454746427671370568340195448570911966902998629125654163 (mod 3713)
  = 232

Efficient alternative (repeated squaring).

- 2003^{1} (mod 3713) = 2003
- 2003^{2} (mod 3713) = 4,012,009 (mod 3713) = 1969
- 2003^{4} (mod 3713) = 1969^{2} (mod 3713) = 589
- 2003^{8} (mod 3713) = 589^{2} (mod 3713) = 1612
- 2003^{16} (mod 3713) = 3157

RSA Details

How large should n = pq be?

- 1,024 bits for long term security.
- Too small $\Rightarrow$ easy to break.
- Too large $\Rightarrow$ time consuming to encrypt/decrypt.

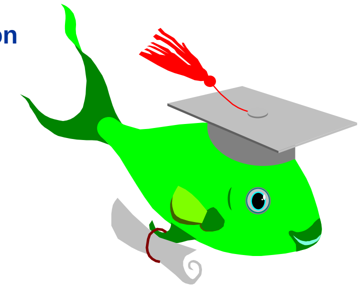
How to choose large "random" prime numbers?

- Miller-Rabin procedure checks whether x is prime. Usually!
  - ✎ Guess, and use subroutine to check.
- Number theory $\Rightarrow$ $n / \log_e n$ prime numbers between 2 and n.
  - ✎ Primes are plentiful: 4.3×10^{97} with $\leq$ 100 digits.

How to compute d efficiently?

- Existence guaranteed since $\gcd(e, \phi) = 1$.
- Fancy version of Euclid's algorithm.

Where to go from Here?



COS 415:	Applied Discrete Optimization
COS 451:	Computational Geometry
COS 487:	Theory of Computation
COS 496:	Cryptography
COS 521:	Advanced Algorithms
COS 524:	Combinatorial Optimization
COS 525:	Mathematical Analysis of Algorithms
COS 528:	Data Structures and Graph Algorithms
COS 551:	Genomics and Computational Biology
ORF 307, 522:	Linear Programming
ORF 547:	Dynamic Programming

Course Evaluations

Course:	COS 423
Instructor:	Kevin Wayne
TAs:	Edith Elkind, Sumeet Sobti
Lecture:	1
Time:	MW 1:30-2:50

Fill out with a #2 pencil:

- Section I: Lectures.
- Section VI: Readings.
- Section VII: Papers, reports, problem sets, examinations.
- Section VIII: General.

All answers are confidential.

Extra Slides

RSA Public-Key Cryptosystem

Why does it work? Rigorously.

- $M' = C^d \pmod n$
  $= M^{ed} \pmod n$

Now, since $\phi = (p-1)(q-1)$ and $e\,d \equiv 1 \pmod \phi$

- $ed = 1 + k(p-1)(q-1)$ for some integer k.

A little manipulation.

- $M^{ed} \equiv M\,M^{(p-1)\,k(q-1)} \pmod p$
  $\equiv M\,(1)^{k(q-1)} \pmod p$
  $\equiv M \pmod p$
  (trivially true if $M \equiv 0$)
- $M^{ed} \equiv M \pmod q$

Finally.

- $M^{ed} \equiv M \quad (\text{mod } \underbrace{pq}_{n})$

Fermat's Little Theorem

if p is prime, then for all $a \neq 0$
$a^{p-1} \equiv 1 \pmod p$

Chinese Remainder Theorem

if p, q prime then for all x, a
$x \equiv a \pmod{pq} \Leftrightarrow$
$x \equiv a \pmod p,\ x \equiv a \pmod q$