Problem. Given a composite integer $n > 1$, find some nontrivial factorization of n, that is $n = uv$ where each of u and v is an integer > 1.

Note: One can be pretty confident about whether a large integer n is composite without knowing a nontrivial factorization.

Expectation. A random number n will have around $\log\log n$ prime factors.

Theorem. *If $\omega(n)$ is the number of distinct prime factors of n, then*

$$\sum_{n\le x}\big(\omega(n)-\log\log x\big)^2 \ll x\log\log x.$$

Corollary. *For almost all n, we have*

$$(*) \qquad |\omega(n)-\log\log n| \le (\log\log n)^{2/3}.$$

Expectation 2. “Most” numbers n have a prime factor $> \sqrt{n}$.

$$\sum_{p \le x} \frac{1}{p} = \log\log x + A + O(1/\log x)$$

$$\sum_{n \le x} \sum_{\substack{\sqrt{x} < p \le x \\ p | n}} 1 = \sum_{\sqrt{x} < p \le x} \sum_{\substack{n \le x \\ p | n}} 1 \ge (\log 2)x + O\left(\frac{x}{\log x}\right)$$

$$\log 2 = 0.69314718\ldots$$

Comment: A random number n will have small prime factors, so it is reasonable to first do a quick "sieve" to determine if this is the case.

How many integers $n \le x$ do not have a prime factor $\le z$?

$$\textbf{On the order of } \frac{x}{\log z}.$$

Pollard's ρ-Algorithm

This method typically finds a prime factor p of n in about $\sqrt{p}$ steps (so $O(n^{1/4})$ steps), and small prime factors of n will usually be found first.

A couple of relevant asides:

The birthday problem and a card trick.

Pollard's ρ-Algorithm

Idea with a hiccup:

- Take $f(x) = x^2 + 1$, and define $f^{(1)}(x) = f(x)$ and $f^{(j+1)}(x) = f(f^{(j)}(x))$ for $j \geq 1$.
- Compute $a_j = f^{(j)}(1) \mod n$ for $1 \leq j \leq k$ where $k \approx \sqrt[4]{n}$ (or less).
- Compute $\gcd(a_i - a_j, n)$ for $1 \leq i < j \leq k$ to get a likely factorization of n.

Why does this likely lead to a factorization of n?

What's the hiccup?

Pollard's ρ-Algorithm

Fixing the difficulty:

Pollard's ρ-Algorithm

Fixing the difficulty: Observe that if $a_i \equiv a_j \pmod{p}$ for a prime factor p of n, then $a_{i+u} \equiv a_{j+u} \pmod{p}$ $\forall\, u \in \mathbb{Z}^+$. Also, there is a $u \in \{1, 2, \dots, j-i\}$ for which $(j-i)|(i+u)$. If $t = i + u$, we get

$$a_t \equiv a_{t+(j-i)} \equiv a_{t+2(j-i)} \equiv a_{t+3(j-i)} \equiv \cdots \equiv a_{2t} \pmod{p}.$$

Compute $a_1, a_2, \dots$ modulo n, and check as one progresses the values of $\gcd(a_{2t} - a_t \bmod n, n)$ for $t = 1, 2, \dots, k$.

Pollard's ρ-Algorithm

- Take $f(x) = x^2 + 1$, and define $f^{(1)}(x) = f(x)$ and $f^{(j+1)}(x) = f(f^{(j)}(x))$ for $j \geq 1$.

- Compute $a_j = f^{(j)}(1) \mod n$ for $1 \leq j \leq k$ where $k \approx \sqrt[4]{n}$ (or less).

- Compute $\gcd(a_i - a_j, n)$ for $1 \leq i < j \leq k$ to get a likely factorization of n.

Compute $a_1, a_2, \dots$ modulo n, and check as one progresses the values of $\gcd(a_{2t} - a_t \bmod n, n)$ for $t = 1, 2, \dots, k$.

Comment: In 1981, Brent and Pollard factored $F_8 = 2^{2^8} + 1$ (containing 78 digits) using this method with $f(x) = x^{1024}+1$.

smallest prime divisor 1238926361552897

Why $f(x) = x^{1024} + 1$?

Dixon's Factoring Algorithm

Basic (Important) Idea (Not Just For Dixon's Algorithm)

- Suppose

  $$n = p_1^{e_1} p_2^{e_2} \cdots p_r^{e_r}$$

  with p_j "odd" distinct primes and $e_j \in \mathbb{Z}^+$.

- Then $x^2 \equiv 1 \pmod{p_j^{e_j}}$ has two solutions which implies $x^2 \equiv 1 \pmod{n}$ has 2^r solutions.

- If x and y are random and $x^2 \equiv y^2 \pmod{n}$, then with probability $(2^r - 2)/2^r$ we can factor n (nontrivially) by considering $\gcd(x + y, n)$.

Dixon's Factoring Algorithm

1. Randomly choose a number $a > \sqrt{n}$ and compute $s(a) = a^2 \mod n$.

2. A bound $B = B(n)$ is chosen (specified momentarily). Determine if $s(a)$ has a prime factor $> B$. We choose a new a if it does. Otherwise, we obtain a complete factorization of $s(a)$.

3. Let $p_1, \ldots, p_t$ denote the primes $\le B$. We continue steps (1) and (2) until we obtain $t + 1$ different a's, say $a_1, \ldots, a_{t+1}$.

4. From the above, we have the factorizations

$$s(a_i) = p_1^{e(i,1)} p_2^{e(i,2)} \cdots p_t^{e(i,t)} \quad \text{for } i \in \{1, 2, \ldots, t+1\}.$$

Dixon's Factoring Algorithm

4. From the above, we have the factorizations

$$s(a_i) = p_1^{e(i,1)} p_2^{e(i,2)} \cdots p_t^{e(i,t)} \quad \text{for } i \in \{1, 2, \ldots, t+1\}.$$

For $i \in \{1, 2, \ldots, t+1\}$, compute the vectors

$$\vec{v}_i = \langle e(i,1), e(i,2), \ldots, e(i,t) \rangle \quad \text{mod } 2.$$

These vectors are linearly dependent modulo 2. Use Gaussian elimination (or something better) to find a non-empty set $S \subseteq \{1, 2, \ldots, t+1\}$ such that $\sum_{i \in S} \vec{v}_i \equiv \vec{0} \pmod{2}$. Calculate $x \in [0, n-1] \cap \mathbb{Z}$ (in an obvious way) satisfying

$$\prod_{i \in S} s(a_i) \equiv x^2 \pmod{n}.$$

4. From the above, we have the factorizations

$$s(a_i) = p_1^{e(i,1)} p_2^{e(i,2)} \cdots p_t^{e(i,t)} \quad \text{for } i \in \{1, 2, \ldots, t+1\}.$$

For $i \in \{1, 2, \ldots, t+1\}$, compute the vectors

$$\vec{v}_i = \langle e(i,1), e(i,2), \ldots, e(i,t) \rangle \mod 2.$$

These vectors are linearly dependent modulo 2. Use Gaussian elimination (or something better) to find a non-empty set $S \subseteq \{1, 2, \ldots, t+1\}$ such that $\sum_{i \in S} \vec{v}_i \equiv \vec{0} \pmod 2$. Calculate $x \in [0, n-1] \cap \mathbb{Z}$ (in an obvious way) satisfying

$$\prod_{i \in S} s(a_i) \equiv x^2 \pmod n.$$

5. Calculate $y = \prod_{i \in S} a_i \mod n$. Then $x^2 \equiv y^2 \pmod n$. Compute $\gcd(x + y, n)$. Hopefully, a nontrivial factorization of n results.

Problem from Old Comprehensive Exam

Given $n = 12371$, describe precisely how to use Dixon's Factoring Algorithm and the following information to find a nontrivial factor of n. You do not need to come up with a factor of n, but use Dixon's Factoring Algorithm to reduce coming up with a factor of n to the computation of $\gcd(a, n)$ where you tell me rather precisely what the value of a is (it should involve multiplication and addition of specific numbers). It is possible that the a you choose will not produce a factorization of n; in the algorithm one might need to try more than one value of a. You need only give me one reasonable choice for a. Use the following information where all congruences shown are modulo n:

$$116^2 \equiv 5 \times 7 \times 31, \quad 136^2 \equiv 5^3 \times 7^2, \quad 159^2 \equiv 7^2 \times 11,$$

$$170^2 \equiv 2 \times 3^3 \times 7 \times 11, \quad 173^2 \equiv 3 \times 7 \times 13 \times 19, \quad 184^2 \equiv 2 \times 3 \times 7^2 \times 31.$$

Small Example: $n = 1189$ and $B = 11$.

Homework: (due October 25 by class time)
page 14, problem (1) about (1) on page 12
page 16 on Dixon's Factoring Algorithm
New Problem below (not in Notes)

Use Dixon's Algorithm to factor $n = 80099$. Suppose $B = 15$ and the a_j's from the first three steps are the numbers 1392, 58360, 27258, 39429, 12556, 42032, and 1234. (Each of these squared mod n should have all of its prime factors $\leq B$.)

Small Example: $n = 1189$ and $B = 11$.

Homework: (due October 25 by class time)
page 14, problem (1) about (1) on page 12
page 16 on Dixon's Factoring Algorithm
New Problem below (not in Notes)

New Problem.

(a) Calculate accurate to 4 decimal places the value of

$$\lim_{x\to\infty} \frac{|\{n \le x : \forall \text{ primes } p \text{ dividing } n, \text{ we have } p \le x^{1/3}\}|}{x}.$$

(b) Calculate accurate to 4 decimal places the value $a \in (0,1)$ such that

$$\lim_{x\to\infty} \frac{|\{n \le x : \forall \text{ primes } p \text{ dividing } n, \text{ we have } p \le x^{a}\}|}{x} = \frac{1}{2}.$$

Small Example: $n = 1189$ and $B = 11$.

Homework: (due October 26 by class time)
page 14, problem (1) about (1) on page 12
page 16 on Dixon's Factoring Algorithm
New Problem below (not in Notes)

Use Dixon's Algorithm to factor $n = 80099$. Suppose $B = 15$ and the a_j's from the first three steps are the numbers 1392, 58360, 27258, 39429, 12556, 42032, and 1234. (Each of these squared mod n should have all of its prime factors $\leq B$.)

MAPLE EXAMPLE