The Blum-Blum-Shub PRG is an example of a PRG, which is believed to be unpredictable.

More precisely, it has been shown that the ability to predict its values is equivalent to being able to efficiently solve the quadratic residuosity problem (which is believed to be hard). Currently, the best way to "solve" the quadratic residuosity problem mod M relies on factoring M. However, factoring large numbers is considered to be hard (and lots of crypto relies on that).

Quadratic residuosity problem. Given big $M=pq$ and a residue x modulo M, decide whether x is a quadratic residue. (About $M/4$ are quadratic residues (the exact number is $\phi(M)/4=(p-1)(q-1)/4$); $M/2$ are easily determined to be nonsquare using the Jacobi symbol [don't worry if you haven't heard about that].)

(Blum-Blum-Shub PRG) Let $M=pq$ where p, q are large primes $\equiv 3 \pmod 4$.
From the seed y_0, we generate $y_{n+1} \equiv y_n^2 \pmod M$.
The random bits x_n we produce are $y_n \pmod 2$ (i.e. $x_n = \text{least bit of}(y_n)$).

B-B-S is very slow, and mostly of theoretical value. However, it is interesting because it is indeed unpredictable (to anyone not knowing the factorization of M) if an important number theory problem is "hard" (this can be made precise), as is believed to be the case.

Why the conditions on p and q? Recall from the CRT that an invertible quadratic residue x^2 modulo $M=pq$ has exactly four squareroots $\pm x$, $\pm y$. The condition $3 \pmod 4$ guarantees that, of these four, exactly one is itself a quadratic residue. As a consequence, the mapping $y \mapsto y^2 \pmod M$ is 1-1 when restricted to invertible quadratic residues (see below).

Comment. For obvious reasons, the seed $y_0 \equiv \pm 1 \pmod M$ should be excluded. Also, for the above considerations, the seed needs to be coprime to M. However, we don't need to worry about that: running into a factor of M by accident is close to impossible (recall that nobody should be able to factor M even on purpose and with lots of time and resources).

Comment. To increase speed, at the expense of some security, we can also take several, say k, bits of y_n (as long as k is small, say, $k \leqslant \log_2 \log_2 M$).

Example 84.

(a) List all invertible quadratic residues modulo 21. Compute the square of all these residues.

(b) Repeat the first part modulo 33 and modulo 35. When computing the squares of these, do you notice a difference modulo 35?

[Note that $35=5\cdot 7$ with $5 \equiv 1 \pmod 4$. This case is excluded in the B-B-S PRG.]

Solution. (final answers only)

(a) Among the $\phi(21)=12$ many invertible elements, the squares are $1, 4, 16$ (exactly a quarter).
Computing the squares: $1^2 \equiv 1$, $4^2 \equiv 16$, $16^2 \equiv 4 \pmod{21}$. Note that the squares are all different!

(b) Modulo 33: among the $\phi(33)=20$ many invertible elements, the squares are $1, 4, 16, 25, 31 \equiv 8^2$ (exactly a quarter). Computing the squares: $1^2 \equiv 1$, $4^2 \equiv 16$, $16^2 \equiv 25$, $25^2 \equiv 31$, $31^2 \equiv 4 \pmod{33}$. Again, all the squares are different!

Modulo 35: among the $\phi(35) = 24$ many invertible elements, the squares are 1, 4, 9, $11 \equiv 9^2$, 16, $29 \equiv 8^2$ (exactly a quarter). Computing the squares: $1^2 \equiv 1$, $4^2 \equiv 16$, $9^2 \equiv 11$, $11^2 \equiv 16$, $16^2 \equiv 11$, $29^2 \equiv 1 \pmod{35}$. Observe that these are not all different: for instance, $9^2 \equiv 16^2 \pmod{35}$.

Advanced comment. The map $x \mapsto x^2 \pmod p$ restricted to invertible quadratic residues is 1-1 if and only if -1 is not a quadratic residue (which, by the next result, is equivalent to $p \equiv 3 \pmod 4$).

[*Sketch of proof.* The map is 1-1 if and only if, for each invertible quadratic residue x^2, exactly one of the two square roots $\pm x$ is itself a quadratic residue. This is equivalent to -1 not being a quadratic residue.

Indeed, if -1 is a quadratic residue, then x and $-x$ are either both quadratic residues or both not.

On the other hand, if not exactly one of $\pm x$ is a quadratic residue then, because exactly half of the invertible residues are quadratic, there would be some pair of residues $\pm z$ which are both quadratic. But then $-zz^{-1} \equiv -1$ would be a quadratic residue.]

Armin Straub
straub@southalabama.edu

Theorem 85. -1 is a quadratic residue modulo (an odd prime) p if and only if $p \equiv 1 \pmod 4$.

In other words, the quadratic congruence $x^2 \equiv -1 \pmod p$ has a solution if and only if $p \equiv 1 \pmod 4$.

Solution. Let us first see that $p \equiv 1 \pmod 4$ is necessary. Assume $x^2 \equiv -1 \pmod p$. Then, by Fermat's little theorem, $x^{p-1} \equiv 1 \pmod p$. On the other hand, $x^{p-1} = (x^2)^{(p-1)/2} \equiv (-1)^{(p-1)/2} \pmod p$. We therefore need $(-1)^{(p-1)/2} = 1$, which is equivalent to $(p-1)/2$ being even. Which is equivalent to $p \equiv 1 \pmod 4$. (Make sure that's absolutely clear!)

On the other hand, assume that $p \equiv 1 \pmod 4$. We will show that $x = \left(\frac{p-1}{2}\right)!$ has the property that $x^2 \equiv -1 \pmod p$. Indeed,

$$\left[\left(\frac{p-1}{2}\right)!\right]^2 = (-1)^{(p-1)/2}\left(1 \cdot 2 \cdot \ldots \cdot \frac{p-1}{2}\right)^2 = (\pm 1) \cdot (\pm 2) \cdot \ldots \cdot \left(\pm \frac{p-1}{2}\right) \equiv -1 \pmod p.$$

[Here, $(\pm 1) \cdot (\pm 2) \cdots$ is short for $1 \cdot (-1) \cdot 2 \cdot (-2) \cdots$.] For the final congruence, observe that $\pm 1, \pm 2, \ldots, \pm \frac{p-1}{2}$ is a complete set of all nonzero residues. When multiplying all residues, each will cancel with its (modular) inverse, except the ones that are their own inverse. But $a \cdot a \equiv 1 \pmod p$ has only the solution $a \equiv \pm 1$, so that ± 1 are the only residues not canceling.

Comment. The final step of our argument is known as Wilson's congruence: $(p-1)! \equiv -1 \pmod p$.

Theorem 86. (advanced) Let $M = pq$ where p, q are primes $\equiv 3 \pmod 4$. Then the sequence generated by $y_{n+1} \equiv y_n^2 \pmod M$ repeats with period dividing $\operatorname{lcm}(\phi(p-1), \phi(q-1))$.

In particular, the period of the corresponding B-B-S PRG divides $\operatorname{lcm}(\phi(p-1), \phi(q-1))$.

Proof.

- Observe that the numbers are $y_n = y_{n-1}^2 = y_{n-2}^4 = \ldots = y_0^{2^n} \pmod M$. Hence, $y_n \equiv y_0^{2^n} \pmod M$.
- Instead of determining the period directly modulo $M = pq$, we determine the periods modulo p and q. [Why? By the CRT, $y_m \equiv y_n \pmod M$ if and only if $y_m \equiv y_n \pmod p$ and $y_m \equiv y_n \pmod q$.] The period modulo M then is the lcm of of the two periods modulo p and q.
- $y_m \equiv y_n \pmod p$
  $\iff$ $y_0^{2^m} \equiv y_0^{2^n} \pmod p$
  $\impliedby$ $2^m \equiv 2^n \pmod{\phi(p)}$
  [it would be "$\iff$" with $2^m \equiv 2^n \pmod k$ where k is the order of $y_0 \pmod p$]
  $\iff$ $2^m \equiv 2^n \pmod{p-1}$
  [note that 2 is not invertible $\pmod{p-1}$; but 2 is invertible $\left(\bmod \frac{p-1}{2}\right)$ because $p \equiv 3 \pmod 4$]
  $\iff$ $2^{m-1} \equiv 2^{n-1} \left(\bmod \frac{p-1}{2}\right)$ [note that $m, n \geqslant 1$]
  $\impliedby$ $m \equiv n \left(\bmod \phi\left(\frac{p-1}{2}\right)\right)$
  [again, it would be "$\iff$" with $m \equiv n \pmod k$ where k is the order of $2 \left(\bmod \frac{p-1}{2}\right)$]
- In other words, the period $m - n$ modulo p divides $\phi\left(\frac{p-1}{2}\right) = \phi(p-1)$.

  Comment. If $p \equiv 3 \pmod 4$, then $\phi\left(\frac{p-1}{2}\right) = \phi(p-1)$. Indeed, note that $p-1$ is divisible by 2 but not by 4. Hence, 2 and $\frac{p-1}{2}$ are coprime, so that $\phi(p-1) = \phi(2)\phi\left(\frac{p-1}{2}\right) = \phi\left(\frac{p-1}{2}\right)$.
- By the CRT, the period modulo $M = pq$ divides $\operatorname{lcm}(\phi(p-1), \phi(q-1))$. □

Example. In Example 83, we had $M = 7 \cdot 11$, so that the period of the PRG must divide $\operatorname{lcm}(\phi(6), \phi(10)) = \operatorname{lcm}(2, 4) = 4$.

Comment. In practice, people therefore say that, for the cycle length of B-B-S to be large, $\gcd(\phi(p-1), \phi(q-1))$ should be small.

Example 87. We mentioned that the unpredictability of the B-B-S PRG relies on the difficulty of factoring large numbers. Here's an indication how difficult it seems to be. In 1991, RSA Laboratories challenged everyone to factor several numbers including:

```
13506641086599522334960321627880596993888147560566702752448514385152651\
06048595338339402871505719094417982072821644715513736804197039641917 4\
30464965892742562393410208643832021103729587257623585096431105640735 0\
15081875106765946292055636855294752135008528794163773285339061097505443\
34999811150056977236890927563
```

Since then, nobody has been able to factor this 1024 bit number (309 decimal digits). Until 2007, cash prizes were offered up to 200,000 USD, with 100,000 USD for the number above.

https://en.wikipedia.org/wiki/RSA_Factoring_Challenge

Let us illustrate how to actually use this number in the B-B-S PRG.

```
Sage] rsa = Integer("13506641086599522334960321627880596993888147560566702752448514385152651\
                     06048595338339402871505719094417982072821644715513736804197039641917 4\
                     30464965892742562393410208643832021103729587257623585096431105640735 0\
                     15081875106765946292055636855294752135008528794163773285339061097505443\
                     34999811150056977236890927563")

Sage] seed = randint(2,rsa-2)

Sage] y = seed; prg = []

Sage] for i in [1..25]:
          y = power_mod(y, 2, rsa)
          prg.append(y % 2)

Sage] prg
```

$$[0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1]$$

If you are able, even after a gigabyte of pseudorandom bits, to predict the next bits with an accuracy better than 50% (which is just pure guessing), then you likely have a shot at factoring the big integer. You would be the first!

Of course, it is not impressive to see a few random bits in the example above. After all, the seed (which you don't know!) itself consists of 1024 random bits. The whole point is that we can, from these 1024 random bits, produce gigabytes of further pseudorandom bits. As of this day, no one would be able to distinguish these from truly random bits.

While all of this works nicely, B-B-S is considered to be too slow for most practical purposes.

Comment. Note that $M = 135...563 \equiv 3 \pmod 4$. Hence it cannot be a product of primes p, q which are both $3 \pmod 4$.

Example 88. (extra) Generate random bits using the B-B-S PRG with $M = 209$ and seed 10. What is the period of the generated sequence? (Then repeat with seed 25.)

Solution. (final answer only) The seed 10 produces the sequence $0, 1, 0, 1, 1, 1, \ldots$ of period 6.

The seed 25 generates the sequence $1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, \ldots$ of period 12.

[By the way, it is an excellent idea to let Sage assist you.]

Primality testing

Recall that it is extremely difficult to factor large integers (this is the starting point for many cryptosystems). Surprisingly, it is much simpler to tell if a number is prime.

Example 89. The following is the number from Example 87, for which RSA Laboratories, until 2007, offered $100,000 to the first one to factorize it. Nobody has been able to do so to this day.

Has the thought crossed your mind that the challengers might be tricking everybody by choosing M to be a huge prime that cannot be factored further? Well, we'll talk more about primality testing soon. But we can actually quickly convince ourselves that M cannot be a prime. If M was prime then, by Fermat's little theorem, $2^{M-1} \equiv 1 \pmod{M}$. Below, we compute $2^{M-1} \pmod{M}$ and find that $2^{M-1} \not\equiv 1 \pmod{M}$. This proves that M is not a prime. It doesn't bring us any closer to factoring it though.

Comment. Ponder this for a while. We can tell that a number is composite without finding its factors. Both sides to this story (first, being able to efficiently tell whether a number is prime, and second, not being able to factor large numbers) are of vital importance to modern cryptography.

```
Sage] rsa = Integer("135066410865995223349603216278805969938881475605667027524485143851\
                     525510604859533833940287150571909441798207282164471551373680419703\
                     964191743046496589274256239341020864383202110372958725762358509643\
                     110564073501508187510676594629205563685529475213500852879416377328\
                     533906109750544334999811150056977236890927563")

Sage] power_mod(2, rsa-1, rsa)

    120939094432033615867650595352956996867540098463588951238902808367556733932202059338 53\
    348534147116662841968124107288512373904071077139405352848835710498409193003137847878 95\
    226029615123284879513798127406300472693925500331497519103479951096634123177725212482 97\
    950196643140069546889855131459759160570963857373851
```

Comment. Just for giggles, let us emphasize once more the need to compute $2^{N-1} \pmod{N}$ without actually computing 2^{N-1}. Take, for instance, the 1024 bit RSA challenge number $N = 135...563$. In Example 89, we computed $2^{N-1} \pmod{N}$, observed that it was $\not\equiv 1$ and concluded that N is not prime. The number 2^{N-1} itself has $N - 1 \approx 2^{1024} \approx 10^{308.3}$ binary digits. It is often quoted that the number of particles in the visible universe is estimated to be between 10^{80} and 10^{100}. Whatever these estimates are worth, our number has WAY more digits (!) than that. Good luck writing it out! [Of course, the binary digits are a single 1 followed by all zeros. However, we need to further compute with that!]

Comment. There is nothing special about 2. You could just as well use, say, 3.

Example 90. (bonus challenge) Find the factors of the following number $M = pq$:

```
893202800574373633936083863874693604950799157730735990874355694281082 7\
076151461165069181335366401887650477533577602609343916545431925218633\
751141065095634529703730490829332440131073471416542829240327143 11
```

As indicated in Example 87, this is difficult. Through some sort of espionage, however, you have learned that $\phi(M)$ is:

```
893202800574373633936083863874693604950799157730735990874355694281082 7\
076151461165069181335366401887657264952783386626998307790668498916912 5\
759563757735725786146787680002256288669908402235207462838677975 12
```

In general, if $M = pq$ is a product of two large primes p, q, given $\phi(M)$, how can we factor M?

Send me the factorization, and an explanation how you found it, by next week for a bonus point!

Comment. Even if we don't know the number of prime factors of M (in the above case we know that M is a product of two primes), we can "efficiently" factor M if we know the value of $\phi(M)$.

Armin Straub
straub@southalabama.edu

Example 91. Fermat's little theorem can be stated in the slightly stronger form:

n is a prime $\iff$ $a^{n-1} \equiv 1 \pmod{n}$ for all $a \in \{1, 2, \ldots, n-1\}$

Why? Fermat's little theorem covers the "$\Longrightarrow$" part. The "$\Longleftarrow$" part is a direct consequence of the fact that, if n is composite with divisor d, then $d^{n-1} \not\equiv 1 \pmod{n}$. (Why?!)

Fermat primality test

Input: number n and parameter k indicating the number of tests to run

Output: "not prime" or "likely prime"

Algorithm:

Repeat k times:
  Pick a random number a from $\{2, 3, \ldots, n-2\}$.
  If $a^{n-1} \not\equiv 1 \pmod{n}$, then stop and output "not prime".
Output "likely prime".

If $a^{n-1} \equiv 1 \pmod{n}$ although n is composite, then a is called a **Fermat liar** modulo n.

On the other hand, if $a^{n-1} \not\equiv 1 \pmod{n}$, then n is composite and a is called a **Fermat witness** modulo n.

Flaw. There exist certain composite numbers n (see Definition 93) for which every a is a Fermat liar (or reveals a factor of n). For this reason, the Fermat primality test should not be used as a general test for primality. That being said, for very large random numbers, it is exceedingly unlikely to meet one of these troublesome numbers, and so the Fermat test is indeed used for the purpose of randomly generating huge primes (for instance in PGP). In fact, in that case, we can even always choose $a = 2$ and $k = 1$ with virtual certainty of not messing up.

Next class, we will discuss an extension of the Fermat primality test which solves these issues (and is just mildly slower).

Advanced comment. If n is composite but not an absolute pseudoprime (see Definition 93), then at least half of the values for a satisfy $a^{n-1} \not\equiv 1 \pmod{n}$ and so reveal that n is not a prime. This is more of a theoretical result: for most large composite n, almost every a (not just half) will be a Fermat witness.

Example 92. Suppose we want to determine whether $n = 221$ is a prime. Simulate the Fermat primality test for the choices $a = 38$ and $a = 24$.

Solution.

- First, maybe we pick $a = 38$ randomly from $\{2, 3, \ldots, 219\}$.
  We then calculate that $38^{220} \equiv 1 \pmod{221}$. So far, 221 is behaving like a prime.
- Next, we might pick $a = 24$ randomly from $\{2, 3, \ldots, 219\}$.
  We then calculate that $24^{220} \equiv 81 \not\equiv 1 \pmod{221}$. We stop and conclude that 221 is not a prime.

Important comment. We have done so without finding a factor of n. (To wit, $221 = 13 \cdot 17$.)

Comment. Since 38 was giving us a false impression regarding the primality of n, it is called a **Fermat liar** modulo 221. Similarly, we say that 221 is a **pseudoprime** to the base 38.

On the other hand, we say that 24 was a **Fermat witness** modulo 221.

Comment. In this example, we were actually unlucky that our first "random" pick was a Fermat liar: only 14 of the 218 numbers (about 6.4%) are liars. As indicated above, for most large composite numbers, the proportion of liars will be exceedingly small.

Somewhat suprisingly, there exist composite numbers n with the following disturbing property: every residue a is a Fermat liar or $\gcd(a,n)>1$.

This means that the Fermat primality test is unable to distinguish n from a prime, unless the randomly picked number a happens to reveal a factor (namely, $\gcd(a,n)$) of n (which is exceedingly unlikely for large numbers).

[Recall that, for large numbers, we do not know how to find factors even if that was our primary goal.]

Such numbers are called absolute pseudoprimes:

Definition 93. A composite positive integer n is an **absolute pseudoprime** (or Carmichael number) if $a^{n-1}\equiv 1 \pmod n$ holds for each integer a with $\gcd(a,n)=1$.

The first few are $561, 1105, 1729, 2465, \ldots$ (it was only shown in 1994 that there are infinitely many of them).

These are very rare, however: there are 43 absolute pseudoprimes less than 10^6. (Versus $78,498$ primes.)

Example 94. Show that 561 is an absolute pseudoprime.

Solution. We need to show that $a^{560}\equiv 1 \pmod{561}$ for all invertible residues a modulo 561.

Since $561=3\cdot 11\cdot 17$, $a^{560}\equiv 1 \pmod{561}$ is eqivalent to $a^{560}\equiv 1 \pmod p$ for each of $p=3,11,17$.

By Fermat's little theorem, we have $a^2\equiv 1 \pmod 3$, $a^{10}\equiv 1 \pmod{11}$, $a^{16}\equiv 1 \pmod{17}$. Since $2,10,16$ each divide 560, it follows that indeed $a^{560}\equiv 1 \pmod p$ for $p=3,11,17$.

Comment. Korselt's criterion (1899) states that what we just observed in fact characterizes absolute pseudoprimes. Namely, a composite number n is an absolute pseudoprime if and only if n is squarefree, and for all primes p dividing n, we also have $p-1|n-1$.

Comment. Our argument above shows that, in fact, $a^{80}\equiv 1 \pmod{561}$ for all invertible residues a modulo 561.

Theorem 95. (Korselt's Criterion) A composite positive integer n is an absolute pseudoprime if and only if n is squarefree and $(p-1)|(n-1)$ for each prime divisor p of n.

Proof. Here, we will only consider the "if" part (the "only if" part is also not hard to show but the typical proof requires a little more insight into primitive roots than we currently have).

To that end, assume that n is **squarefree** and that $(p-1)|(n-1)$ for each prime divisor p of n. Let a be any integer with $\gcd(a,n)=1$. We will show that $a^{n-1}\equiv 1 \pmod n$.

n being squarefree means that its prime factorization is of the form $n=p_1\cdot p_2\cdots p_d$ for distinct primes p_i (this is equivalent to saying that there is no integer $m>1$ such that $m^2|n$). By Fermat's little theorem $a^{p_i-1}\equiv 1 \pmod{p_i}$ and, since $(p_i-1)|(n-1)$, we have $a^{n-1}\equiv 1 \pmod{p_i}$ for all p_i. It therefore follows from the Chinese remainder theorem that $a^{n-1}\equiv 1 \pmod n$. □

Comment. Modulo a prime p, Fermat's little theorem implies that $a^p\equiv a \pmod p$ for each integer a. (Why?!) It therefore follows from the above argument that, for an absolute pseudoprime n, we have $a^n\equiv a \pmod n$ for each integer a (and this property characterizes absolute pseudoprimes).

Review. If N is composite, then a residue a is a Fermat liar modulo N if $a^{N-1} \equiv 1 \pmod{N}$.

Example 96. Using Sage, determine all numbers n up to 5000, for which 2 is a Fermat liar.

```
Sage] def is_fermat_liar(x, n):
          return not is_prime(n) and power_mod(x, n-1, n) == 1

Sage] [ n for n in [1..5000] if is_fermat_liar(2, n) ]
```

$$[341, 561, 645, 1105, 1387, 1729, 1905, 2047, 2465, 2701, 2821, 3277, 4033, 4369, 4371, 4681]$$

Even if you have never written any code, you can surely figure out what's going on!

Heads-up! The improved primality test discussed today will reduce this list to just $2047, 3277, 4033, 4681$.

The Miller–Rabin primality test

Review. The congruence $x^2 \equiv 1 \pmod{p}$ has only the solutions $x \equiv \pm 1$.

By contrast, if n is composite (and odd), then $x^2 \equiv 1 \pmod{n}$ has additional solutions.

The Miller–Rabin primality test exploits this difference to fix the issues of the Fermat primality test.

The Fermat primality test picks a and checks whether $a^{n-1} \equiv 1 \pmod{n}$.

- If $a^{n-1} \not\equiv 1 \pmod{n}$, then we are done because n is definitely not a prime.
- If $a^{n-1} \equiv 1 \pmod{n}$, then either n is prime or a is a Fermat liar.
  But instead of leaving off here, we can dig a little deeper:
  Note that $a^{(n-1)/2}$ satisfies $x^2 \equiv 1 \pmod{n}$. If n is prime, then $x \equiv \pm 1$ so that $a^{(n-1)/2} \equiv \pm 1 \pmod{n}$.
  - Hence, if $a^{(n-1)/2} \not\equiv \pm 1 \pmod{n}$, then we again know for sure that n is not a prime.
    Advanced comment. In fact, we can now factor n! See bonus challenge below.
  - If $a^{(n-1)/2} \equiv 1 \pmod{n}$ and $\frac{n-1}{2}$ is divisible by 2, we continue and look at $a^{(n-1)/4} \pmod{n}$.
    - If $a^{(n-1)/4} \not\equiv \pm 1 \pmod{n}$, then n is not a prime.
    - If $a^{(n-1)/4} \equiv 1 \pmod{n}$ and $\frac{n-1}{4}$ is divisible by 2, we continue...

Write $n - 1 = 2^s \cdot m$ with m odd. In conclusion, if n is a prime, then

$$a^m \equiv 1 \quad \text{or, for some } r = 0, 1, \ldots, s-1, \quad a^{2^r m} \equiv -1 \pmod{n}.$$

In other words, if n is a prime, then the values $a^m, a^{2m}, \ldots, a^{2^s m}$ must be of the form $1, 1, \ldots, 1$ or $\ldots, -1, 1, 1, \ldots, 1$. If the values are of this form even though n is composite, then a is a **strong liar** modulo n.

This gives rise to the following improved primality test:

Miller–Rabin primality test

Input: number n and parameter k indicating the number of tests to run

Output: "not prime" or "likely prime"

Algorithm:

Write $n - 1 = 2^s \cdot m$ with m odd.
Repeat k times:
  Pick a random number a from $\{2, 3, \ldots, n-2\}$.
  If $a^m \not\equiv 1 \pmod{n}$ and $a^{2^r m} \not\equiv -1 \pmod{n}$ for all $r = 0, 1, \ldots, s-1$, then
    stop and output "not prime".
Output "likely prime".

Comment. If n is composite, then less than a quarter of the values for a can possibly be strong liars. In other words, for all composite numbers, the odds that the Miller–Rabin test returns "likely prime" are less than 4^{-k}.

Comment. Note that, though it looks more involved, the Miller–Rabin test is essentially as fast as the Fermat primality test (recall that, to compute a^{n-1}, we proceed using binary exponentiation).

Advanced comments. This is usually implemented as a probabilistic test. However, assuming GRH (the generalized Riemann hypothesis), it becomes a deterministic algorithm if we check $a = 2, 3, \ldots, \lfloor 2(\log n)^2 \rfloor$. This is mostly of interest for theoretical applications. For instance, this then becomes a polynomial time algorithm for checking whether a number is prime.

More recently, in 2002, the AKS primality test was devised. This test is polynomial time (without relying on outstanding conjectures like GRH).

Example 97. Suppose we want to determine whether $n = 221$ is a prime. Simulate the Miller–Rabin primality test for the choices $a = 24$, $a = 38$ and $a = 47$.

Solution. $n - 1 = 4 \cdot 55 = 2^s \cdot m$ with $s = 2$ and $m = 55$.

- For $a = 24$, we compute $a^m = 24^{55} \equiv 80 \not\equiv \pm 1 \pmod{221}$. We continue with $a^{2m} \equiv 80^2 \equiv 212 \not\equiv -1$, and conclude that n is not a prime.

  Note. We do not actually need to compute that $a^{n-1} = a^{4m} \equiv 81$, which features in the Fermat test and which would also lead us to conclude that n is not prime.

- For $a = 38$, we compute $a^m = 38^{55} \equiv 64 \not\equiv \pm 1 \pmod{221}$. We continue with $a^{2m} \equiv 64^2 \equiv 118 \not\equiv -1$ and conclude that n is not a prime.

  Note. This case is somewhat different from the previous in that 38 is a Fermat liar. Indeed, $a^{4m} \equiv 118^2 \equiv 1 \pmod{221}$. This means that we have found a nontrivial sqareroot of 1. In this case, the Fermat test would have failed us while the Miller–Rabin test succeeds.

- For $a = 47$, we compute $a^m = 47^{55} \equiv 174 \not\equiv \pm 1 \pmod{221}$. We continue with $a^{2m} \equiv 174^2 \equiv -1$. We conclude that n is a prime or a is a strong liar. In other words, we are not sure but are (incorrectly) leaning towards thinking that 221 was likely a prime.

Comment. In this example, only 4 of the 218 residues $2, 3, \ldots, 219$ are strong liars (namely $21, 47, 174, 200$). For comparison, there are 14 Fermat liars (namely $18, 21, 38, 47, 64, 86, 103, 118, 135, 157, 174, 183, 200, 203$).

Example 98. In Example 94, we saw that all $\phi(561) = 320$ invertible residues a modulo 561 are Fermat liars (that is, they all satisfy $a^{560} \equiv 1 \pmod{561}$). How many of them are strong liars?

Solution. Only 8 of the 558 residues $2, 3, \ldots, 559$ are strong liars (namely $50, 101, 103, 256, 305, 458, 460, 511$). That's about 1.43% (much less than the theoretic bound of 25%).

(bonus challenge) For which $N < 1000$ is the proportion of strong liars the highest?

Here (as illustrated in the case of 561 above) we define the proportion of strong liars to be the proportion of residues among $2, 3, \ldots, N-2$, which are strong liars.

[That proportion is almost 23%, just shy of the theoretical bound of 25%.]

Send in a solution by next week for a bonus point!

Example 99. How can you check whether a huge randomly selected number N is prime?

Solution. Compute $2^{N-1} \pmod N$ using binary exponentiation. If this is $\not\equiv 1 \pmod N$, then N is not a prime. Otherwise, N is a prime or 2 is a Fermat liar modulo N (but the latter is exceedingly unlikely for a huge randomly selected number N; the bonus challenge below indicates that this is almost as unlikely as randomly running into a factor of N).

Comment. There is nothing special about 2 here (you could also choose 3 or any other generic residue).

Example 100. (bonus challenge) If $a^{n-1} \equiv 1 \pmod n$ but $a^{(n-1)/2} \not\equiv \pm 1 \pmod n$, then we can find a factor of n! How?!

For instance. $a = 38$ and $n = 221$ in Example 97.

Comment. However, note that this only happens if a is a Fermat liar modulo n, and these are typically very rare. So, unfortunately, we have not discovered an efficient factorization algorithm. [But we have run into an idea, which is used for some of the best known factorization algorithms. If time permits, more on that later...]

Send in a solution by next week for a bonus point!

How many primes are there?

Theorem 101. (Euclid) There are infinitely many primes.

Proof. Assume (for contradiction) there are only finitely many primes: $p_1, p_2, \ldots, p_n$.
Consider the number $N = p_1 \cdot p_2 \cdot \ldots \cdot p_n + 1$.
None of the p_i divide N (because division of N by any p_i leaves remainder 1).
Thus any prime dividing N is not on our list. Contradiction.

Just being silly. Similarly, there are infinitely many composite numbers.

Indeed, assume (for contradiction) there are only finitely many composites: $m_1, m_2, \ldots, m_n$.
Consider the number $N = m_1 \cdot m_2 \cdot \ldots \cdot m_n$ (don't add 1).
N is not on our list. Contradiction.

Historical note. This is not necessarily a proof by contradiction, and Euclid (300BC) himself didn't state it as such. Instead, one can think of it as a constructive machinery of producing more primes, starting from any finite collection of primes. □

The following famous and deep result quantifies the infinitude of primes.

Theorem 102. (prime number theorem) Let $\pi(x)$ be the number of primes $\leqslant x$. Then

$$\lim_{x \to \infty} \frac{\pi(x)}{x/\ln(x)} = 1.$$

In other words: Up to x, there are roughly $x/\ln(x)$ many primes.

Examples.

proportion of primes up to 10^6: $\frac{78,498}{10^6} = 7.85\%$ vs the estimate $\frac{1}{\ln(10^6)} = \frac{1}{6\ln(10)} = 7.24\%$

proportion of primes up to 10^{12}: $\frac{37,607,912,018}{10^{12}} = 3.76\%$ vs the estimate $\frac{1}{\ln(10^{12})} = \frac{1}{12\ln(10)} = 3.62\%$

An example of huge relevance for crypto.

By the PNT, the proportion of primes up to 2^{2048} is about $\frac{1}{\ln(2^{2048})} = 0.0704\%$.

That means, roughly, 1 in 1500 numbers of this magnitude are prime. That means we (i.e. our computer) can efficiently generate large random primes by just repeatedly generating large random numbers and discarding those that are not prime.

Comment. Here, $\ln(x)$ is the logarithm with base e. Isn't it wonderful how Euler's number $e \approx 2.71828$ is sneaking up on the primes?

Historical comment. Despite progress by Chebyshev (who succeeded in 1852 in showing that the quotient in the above limit is bounded, for large x, by constants close to 1), the PNT was not proved until 1896 by Hadamard and, independently, de la Vallée Poussin, who both used new ideas due to Riemann.

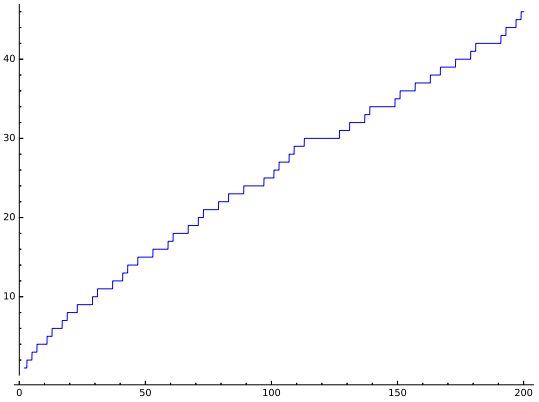
Armin Straub
straub@southalabama.edu

Example 103. Playing with the prime number theorem in Sage:

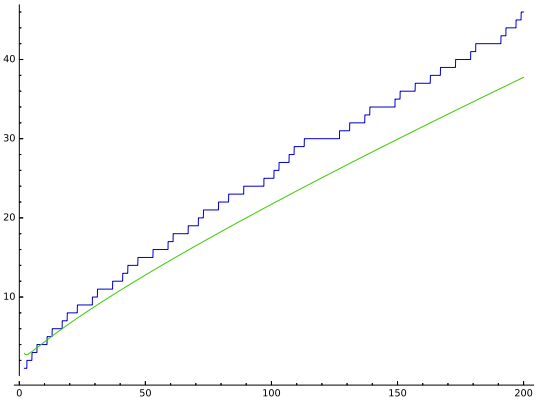
```
Sage] prime_pi(10)
```

```
4
```

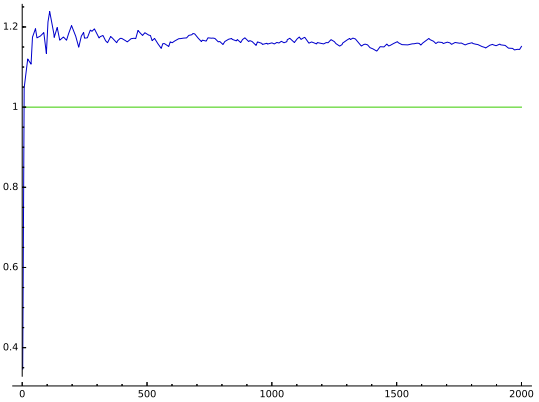
```
Sage] plot(prime_pi(x), 2, 200)
```




```
Sage] plot([prime_pi(x),x/ln(x)], 2, 200)
```




```
Sage] plot([prime_pi(x)/(x/ln(x)), 1], 2, 2000)
```



Comment. As the final plot suggests, the quotient of $\pi(x)$ and $x/\ln(x)$ indeed approaches 1 from above. This is slightly stronger than the PNT, which only claims that the quotient approaches 1.

In particular, as the previous plot suggests, for large x, $x/\ln(x)$ is always an underestimate for $\pi(x)$ (though looking at a plot like this can be very misleading).

Example 104. In 12/2018, a new largest (proven) prime was found: $2^{82,589,933}-1$.

https://www.mersenne.org/primes/?press=M82589933

This is a **Mersenne prime** (like the last 17 record primes). It has a bit over 24.8 million (decimal) digits (versus 23.2 for the previous record). The prime was found as part of GIMPS (Great Internet Mersenne Prime Search), which offers a $3,000 award for each new Mersenne prime discovered.

The EFF (Electronic Frontier Foundation) is offering $150,000 (donated anonymously for that specific purpose) for the discovery of the first prime with at least 100 million decimal digits.

https://www.eff.org/awards/coop

[Prizes of $50,000 and $100,000 for primes with 1 and 10 million digits have been claimed in 2000 and 2009.]

Definition 105. A **Mersenne prime** is a prime of the form 2^n-1.

For instance. The first few Mersenne primes have exponents $2, 3, 5, 7, 13, 17, 19, 31, 61, 89, 107, \ldots$ All of these exponents are primes (but not all primes work: for instance, $2^{11}-1=23\cdot 89$). See below.

Anecdote. Euler proved in 1772 that $2^{31}-1$ is prime (then, and until 1867, the largest known prime).
"$2^{31}-1$ is probably the greatest [Mersenne prime] that ever will be discovered; for as they are merely curious, without being useful, it is not likely that any person will attempt to find one beyond it." — P. Barlow, 1811

https://en.wikipedia.org/wiki/2,147,483,647

Mersenne primes give rise precisely to all even perfect numbers (numbers whose proper divisors sum to the number itself; for instance, 6 is perfect because $6=1+2+3$). Indeed, Euclid showed that, if 2^p-1 is prime, then $2^{p-1}(2^p-1)$ is perfect [$p=2$: $2\cdot 3=6$, $p=3$: $4\cdot 7=28=1+2+4+7+14$, $p=5$: $16\cdot 33=528$, ...]. It is not known whether odd perfect numbers exist.

Example 106. (geometric sum) Evaluate $1+x+x^2+\ldots+x^n$.

Solution. $(1+x+x^2+\ldots+x^n)(x-1)=x^{n+1}-1$, so that $1+x+x^2+\ldots+x^n=\frac{x^{n+1}-1}{x-1}$.

Geometric series. In particular, $\sum_{k=1}^{\infty} x^k=\lim_{n\to\infty}\frac{x^{n+1}-1}{x-1}=\frac{1}{1-x}$, provided that $|x|<1$.

Lemma 107. If $r \mid n$, then $x^r-1 \mid x^n-1$.

Proof. Write $n=rs$. It follows from $x^s-1=(x-1)(1+x+x^2+\ldots+x^{s-1})$ that
$x^{rs}-1=(x^r-1)(1+x^r+x^{2r}+\ldots+x^{r(s-1)})$. □

Corollary 108. 2^n-1 can only be prime if n is prime.

Proof. It follows from the previous lemma that, if $n=rs$ is composite, then 2^n-1 is divisible by 2^r-1 (as well as 2^s-1). □

For instance. $2^6-1=63$ is divisible by both $2^2-1=3$ and $2^3-1=7$.

Armin Straub
straub@southalabama.edu