Algorithm Proof:

Rabin-Miller Theory for Primality

The Rabin-Miller Theory takes advantage of the fact that there are two ways to prove if a number is composite:

n is odd

Put $n - 1 = 2^k q$ (factoring out 2s), where q is odd.
If $a^q \not\equiv 1 \pmod{n}$ and

$a^{2^{(i)}q} \not\equiv -1 \pmod{n}$ for $i = 0, 1, \ldots, k-1$

for some a that is not divisible by n, then n is composite.

If n is composite, then at least 75% of such **a** will be witnesses to a number being composite. The test does not ensure that a number is prime; however, the more **a's** that are tested and fail the two above conditions, then the higher the chance that n is prime.

Proof by Contrapositive:

Let p be an odd prime.

$p - 1 = 2^k q$, where q is odd

$a^q, a^{2q}, a^{2^{(2)}q}, \ldots, a^{2^{(k)}q} \pmod{p}$
$1 \quad p \nmid a$
$\rightarrow \gcd(p, a) = 1$

so, by Fermat's Little Theorem, this is congruent to 1 mod p. Every number in the list is the square of the number before it and the last number is 1 so there are 2 different cases that can happen.

Case 1: the first number is one, then all numbers are ones, in which case $a^q \equiv 1 \bmod n$ and the proof is complete.

Case 2: one number in the list is not one, but eventually the number is squared and becomes one, so consider the equation

$$x^2 \equiv 1 \pmod{p}$$

then there are two solutions to the above equation, x = -1, or x = 1 (mod p). This explains that if there is a number that is not one on the list, then eventually negative one has to show up because when it's squared, it becomes one (Fermat's Little Theorem), but only if these two numbers once squared become 1 mod p. Therefore, if one of these numbers is

not one, then one of them has to be negative one, thus $a^{2^i q} \equiv -1 \pmod{n}$ for some I in the range 0, 1, …, k – 1 and the proof is complete.

The algorithm utilized in the code creates 4 threads, each testing n first against the base cases of even numbers, multiples of 3 and multiples of 5, then against the above two conditions for compositeness. The ‘a’ variable in the above conditions is randomly generated and n is tested against 4 different ‘a’ variables since four threads are created. Thus, there is a higher chance that the number is prime versus composite because it has been tested multiple times against the conditions above, which have been proven to show compositeness in the proof above.

Works Cited

cguy3. “C Program to Implement the Rabin-Miller Primality Test to Check If a Given Number Is Prime.” Sanfoundry, 27 Nov. 2014, www.sanfoundry.com/c-program-implement-rabin-miller-primality-test-check-number-prime/.

Drunk, Derivin, director. YouTube. YouTube, YouTube, 21 Mar. 2014, www.youtube.com/watch?v=SSpcBIM9Gb8.

“Primality Tests.” Number Theory - Primality Tests, http://crypto.stanford.edu/pbc/notes/numbertheory/millerrabin.html.

“Rabin-Miller Primality Test.” Http://Homepages.math.uic.edu, http://homepages.math.uic.edu/~marker/math435/rm.pdf.