Explanation:
Miller Rabin: How to apply?
Check primality of n where $n-1=2^r*t$, t odd
1. Pick random $a>0$
2. Compute $b \equiv a^t \bmod n$ (congruence)
3. If b in $\{-1,1\}$ then "probably prime"
4. For $i=1$ to $r-1$ do
   a) compute $b \equiv b^2 \bmod n$ (assigning to b the new value)
   b) if $b \equiv -1$ output "probably prime"
   c) if $b \equiv 1$ output "n not prime"
5. output "n not prime"

Iterate this for l choices of a to get probablilty of 2^{-l} .

Why does it work?
Fermat says $a^{(n-1)} \equiv a^{(t*2^r)} \equiv 1 \bmod n$ if n is prime
so in the final squaring we need to reach 1 or n is not prime.

If n is prime then there are 2 square roots of 1, namely 1 and -1.
If $n = p * q$ then there are 4 roots, for k different factors there are 2^k roots,
because of CRT.

$x^2 \equiv 1 \bmod n$, let $n = p*q$.
$x^2 \equiv 1 \bmod p$
$x^2 \equiv 1 \bmod q$

p and q are primes, so there 2 squareroots This gives 4 different CRT systems
$x \equiv \pm 1 \bmod p$
$x \equiv \pm 1 \bmod q$
with signs taken independently, these give 4 different solutions, namely
$x \equiv 1 \bmod n$ for both choices +, $x \equiv -1 \bmod n$ if both choices are - and a different solution $x \equiv c \bmod n$ in the case of + for p, - for q and -c in the other.
If we find c with $c^2 \equiv 1 \bmod n$ and c is not +/-1 then n cannot be prime.

Miller Rabin tries to find such c, knowing that $a^{(n-1)} \equiv 1$, and we can compute r square roots of that -- by building the powers of a^t by squaring. If Fermat holds, we must compute 1 eventually, and if n is prime, we must have encountered -1 before that.
This covers the for loop, if a^t is already 1 then we don't get any information.