

Prime Sieve Notes

Lecture link: <https://tegr.it/y/1gkvu>

The prime sieve allows us to efficiently generate a list of primes upto some given integer, n . It works as follows:

- 1) Write down all the numbers from 2 to n .
- 2) Go through each number, in order.
- 3) For each of these, if it's not crossed off, circle it.
- 4) Then, cross off each multiple of that number. Thus, when we circle 2 at the beginning of the algorithm, then cross off 4, 6, 8, 10, and so forth, until we get to the last even number less than or equal to n .

The numbers not crossed off at the end of these (the circled ones) are all the primes in the range.

One note:

If a number, n , is NOT prime, then we know there exist a and b such that $1 < a \leq b$ and

$$n = a \times b$$

In the equality case, we have:

$$n = a \times a = a^2, \text{ so } a = \sqrt{n}$$

This corresponds to the largest value that a could be because if it were any bigger, it would no longer be smaller than b !

Thus, if a number is composite (not prime), then it has at least one factor less than or equal to its square root. So, we can technically stop our outer loop when we get to the square root of the number we are checking.

Secondly, if a number is already crossed off, there is no need to cross off its multiples. For example, when we get to 6, all of its multiples were crossed off when we circled 2, so there's no need to cross them off again.