

Analysis: Number Sets

This problem describes a process where integers are grouped into sets, and then we are asked how many sets there are in the final result.

The process described is fairly slow:

- Create singleton sets of integers.
- Take each pair of integers;
- factor the two integers and see if they have a prime factor in common that is greater than or equal to P ;
- if so, merge the sets that have these two integers.

We need to find the result of this process using a faster method. A few observations help here.

Firstly, we only need to find prime factors less than the size of the interval. If a prime is larger than or equal to the size of the interval, then at most one integer in the interval can have that prime as a factor, so it will never be used to merge sets.

Secondly, we can take a faster approach than finding the prime factors of each integer separately. Instead we consider each prime in turn and find all the integers in the interval which have this prime as a factor, a technique generally called a *sieve*.

Thirdly, joining sets can be implemented efficiently with a data structure called *union-find* (or the *disjoint sets data structure*). As we consider each prime and find all of the integers with that prime as a factor, we join all of the sets containing those integers. We could also build an undirected graph with nodes for the integers in the interval and the primes, and edges representing which integers are divisible by which primes; then the final sets are the connected components of this graph.

More information:

[Prime sieving](#) - [Union-Find](#) - [Connected components](#)