

# There Is No Largest Prime Number

Euclid of Alexandria  
euclid@alexandria.edu  
October 27, 2018

# Outline

## **1. Motivation**

### 1.1. The Basic Problem That We Studied

# 1. Motivation

## 1.1 The Basic Problem That We Studied

# What Are Prime Numbers? I

## ***Definition: Prime number***

A *prime number* is a number that has exactly two divisors.

# What Are Prime Numbers? II

## *Example:*

- 2 is prime (two divisors: 1 and 2).
- 3 is prime (two divisors: 1 and 3).
- 4 is not prime (**three** divisors: 1, 2, and 4).

# *There Is No Largest Prime Number*

## *Theorem: Prime numbers*

There is no largest prime number.

# *There Is No Largest Prime Number*

*Proof:*

1. Suppose  $p$  were the largest prime number.

# *There Is No Largest Prime Number*

## *Proof:*

1. Suppose  $p$  were the largest prime number.
2. Let  $q$  be the product of the first  $p$  numbers.



# *There Is No Largest Prime Number*

## *Proof:*

1. Suppose  $p$  were the largest prime number.
2. Let  $q$  be the product of the first  $p$  numbers.
3. Then  $q + 1$  is not divisible by any of them.

# *There Is No Largest Prime Number*

## *Proof:*

1. Suppose  $p$  were the largest prime number.
2. Let  $q$  be the product of the first  $p$  numbers.
3. Then  $q + 1$  is not divisible by any of them.
4. But  $q + 1$  is greater than 1, thus divisible by some prime number not in the first  $p$  numbers.

The proof used *reductio ad absurdum*.

# *What's Still To Do?*

- Answered Questions
  - How many primes are there?
- Open Questions
  - Is every even number the sum of two primes?

# *An Algorithm For Finding Prime Numbers.*

## *FindPrimeNumbers*

```
int main (void)
{
    std::vector<bool> is_prime (100, true);
    for (int i = 2; i < 100; i++)
        if (is_prime[i])
        {
            std::cout << i << "_";
            for (int j = i; j < 100; is_prime [j] = false, j
                +=i);
        }
    return 0;
}
```

# *An Algorithm For Finding Prime Numbers.*

## *FindPrimeNumbers*

```
int main (void)
{
    std::vector<bool> is_prime (100, true);
    for (int i = 2; i < 100; i++)
        if (is_prime[i])
        {
            std::cout << i << "_";
            for (int j = i; j < 100; is_prime [j] = false, j
                +=i);
        }
    return 0;
}
```

Note the use of std::.

*It's me, Euclid*



Figure: *It's me, Euclid* [1]

# References I

- [1] URL: [https://upload.wikimedia.org/wikipedia/commons/3/30/Euklid-von-Alexandria\\_1.jpg](https://upload.wikimedia.org/wikipedia/commons/3/30/Euklid-von-Alexandria_1.jpg) (visited on 10/22/2018).
- [2] [Noam Chomsky](#). *Syntactic Structures*. The Hague: Mouton, 1957.
- [3] [William Labov](#). *Sociolinguistic Patterns*. Philadelphia: University of Pennsylvania Press, 1972.