

There Is No Largest Prime Number

Long Subtitle

Author¹ Euclid of Alexandria²

¹Department of Computer Science
University of Somewhere

²Department of Philosophy
University of Alexandria

27th International Symposium of Prime Numbers

Outline

- 1 Motivation
 - Changed to something more reasonable
- 2 Results
 - Somethin' else

Outline

- 1 Motivation
 - Changed to something more reasonable
- 2 Results
 - Somethin' else

What Are Prime Numbers?

Definition

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

Example

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

There Is No Largest Prime Number

The proof uses *reductio ad absurdum*.

Use of `uncover` command.

Theorem

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*.

There Is No Largest Prime Number

The proof uses *reductio ad absurdum*.

Use of `uncover` command.

Theorem

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*.

There Is No Largest Prime Number

The proof uses *reductio ad absurdum*.

Use of `uncover` command.

Theorem

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*.

There Is No Largest Prime Number

The proof uses *reductio ad absurdum*.

Use of `uncover` command.

Theorem

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*.

There Is No Largest Prime Number

The proof uses *reductio ad absurdum*.

Use of `only` command.

Theorem

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. □

There Is No Largest Prime Number

The proof uses *reductio ad absurdum*.

Use of `only` command.

Theorem

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. □

There Is No Largest Prime Number

The proof uses *reductio ad absurdum*.

Use of `only` command.

Theorem

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. □

There Is No Largest Prime Number

The proof uses *reductio ad absurdum*.

Use of `only` command.

Theorem

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?

One option:

Answered Questions

How many primes are there?

Open Questions

Is every even number the sum of two primes?

What's Still To Do?

Another option:

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

What's Still To Do?

Yet another option...

Answered Questions

How many primes are there?

Open Questions

Is every even number the sum of two primes? [3]

What's Still To Do?

Yet another option...

Answered Questions

How many primes are there?

Open Questions

Is every even number the sum of two primes? [3]

An Algorithm For Finding Prime Numbers.

```
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::`.

An Algorithm For Finding Prime Numbers.

```
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::`.

An Algorithm For Finding Prime Numbers.

```
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.

```
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.

```
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.

```
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::`.

For Further Reading I



A. Author.

Handbook of Everything.

Some Press, 1990.



S. Someone.

On this and that.

Journal of This and That, 2(1):50–100, 2000.



[Goldbach, 1742] Christian Goldbach.

A problem we should try to solve before the ISPN '43 deadline,

Letter to Leonhard Euler, 1742.