LaTeX-Kurs

Philipp Arras, Florian Nowak

11. Oktober 2014

Folie 1

1. Erstens
2. Zweitens
3. Drittens

There Is No Largest Prime Number

The proof uses *reductio ad absurdum*.

Theorem

There is no largest prime number.

Beweis.

1. Suppose p were the largest prime number.

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*.

Theorem

There is no largest prime number.

Beweis.

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

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*.

Theorem

There is no largest prime number.

Beweis.

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*.

Theorem

There is no largest prime number.

Beweis.

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?

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 Primes Numbers.


```
int main (void)
{
  std::vector<bool> is_prime (100, true);
  for (int i = 2; i < 100; i++)




  return 0;
}
```

An Algorithm For Finding Primes Numbers.


```
int main (void)
{
  std::vector<bool> is_prime (100, true);
  for (int i = 2; i < 100; i++)
if (is_prime[i])
  {



is_prime [j] = false, j+=i);
  return 0;
}
```

An Algorithm For Finding Primes 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 Primes 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::`.