# Proving $\pi$ is not Equal to Various Fractions

Subat Begzad, Suhanee Patel, Charvi Reddy,
Dhruv Srivastava, Sanjana Thalvayapati
Mentor: William Gasarch

August 2026

## 1 Introduction

For thousands of years, mathematicians have tried to find accurate ways to approximate the mathematical constant $\pi$, the ratio of a circle's circumference to its diameter. Since $\pi$ has an infinite, non-repeating decimal, it cannot be written exactly as a fraction, making approximations essential for practical calculations throughout history.

One of the most well-known approximations is 22/7, which has been used for over two thousand years. Although it is commonly attributed to the Greek mathematician Archimedes, who established upper and lower bounds for $\pi$ using inscribed and circumscribed polygons, there is no evidence that he claimed $\pi$ was exactly equal to 22/7. Instead, his work demonstrated that $\pi$ lies between two rational numbers, with 22/7 serving as a convenient upper bound. Centuries later, Chinese mathematician Zu Chongzhi discovered a better approximation, 355/113. It was known as Milü, and it was one of the most accurate fractional approximations to $\pi$ available for nearly one thousand years. Another important approximation, 333/106, also emerged from continued efforts to produce simple fractions that more closely matched the true value of $\pi$ while remaining practical for calculations.

Traditionally, one would prove that none of these fractions equal $\pi$ by showing that $\pi$ is irrational, meaning it cannot be expressed exactly as the ratio of two integers. Incidentally, $\pi$ is also a transcendental number, meaning it is not the solution of any polynomial equation with integer coefficients. However, proving $\pi$ is either of those is rather difficult. We would like to prove that $\pi$ is not equal to these fractions using more elementary means.

## 2 Using Inscribed and Circumscribed Polygons

The first method uses inscribed and circumscribed regular polygons to approximate $\pi$. We proceed by inscribing and circumscribing various regular polygons inside a unit circle. An inscribed polygon lies entirely within the circle, so its perimeter is always less than the circumference. A circumscribed polygon is

necessarily larger than the circle, so its perimeter is always greater than the circumference. Therefore, $PI(n) < \pi < PC(n)$. To find an exact formula, we use elementary trigonometry.

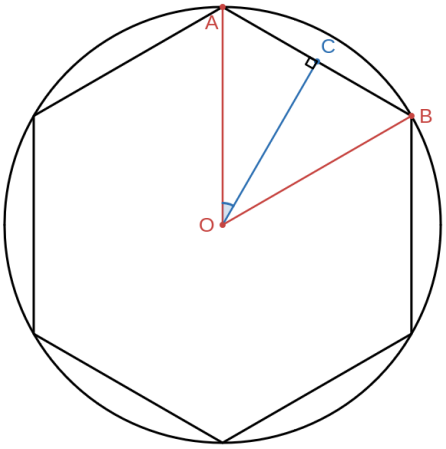



Figure 1: A regular hexagon inscribed in a unit circle.

We draw an altitude $OC$. Since the circle is a unit circle, $OA = 1$. Since the polygon is regular and $OC$ is an angle bisector, $\angle OAC = 2\pi/2n = \pi/n$. By trigonometry, $AC = OA \cdot \sin(\angle OAC) = \sin(\pi/n)$. Because $AC$ is equal to the perimeter of the polygon divided by $2n$, the perimeter is $2n \sin(\pi/n)$. We know that the circumference of the circle is $2\pi$, so $PI(n) = n \sin(\pi/n)$ provides an approximation for $\pi$.

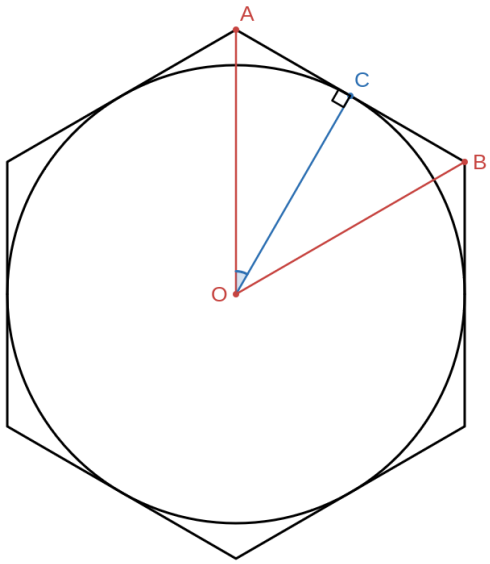



Figure 2: A regular hexagon circumscribed about a unit circle.

For the circumscribed circle, the method is similar. We draw an altitude $OC$. Since $OC$ is a radius of a unit circle, it has length 1. Again, $\angle AOC = \pi/n$. By trigonometry, $AC = OC \cdot \tan(\angle AOC) = \tan(\pi/n)$. As before, this implies the approximation for $\pi$ is $PC(n) = n \tan(\pi/n)$.

To estimate $\pi$, we used the average of these two values:

$$A(n) = \frac{PI(n) + PC(n)}{2}.$$

We used Python to generate the polygon approximations and find the smallest value of $n$ needed to prove that each fraction is not equal to $\pi$. The table shows how the polygon approximation improves as the number of polygon sides increases.

Let $X(n)$ represent the approximate number of correct digits of $\pi$ in $A(n)$. From the data, the approximations have about 2 correct digits when $n = 10$, 4 when $n = 100$, 6 when $n = 1000$, and 8 when $n = 10000$. Each time $n$ is multiplied by 10, the number of correct digits increases by 2. The data suggests that $X(n) \approx 2 \log_{10}(n)$. For example, when $n = 1000$, $X(1000) \approx 2 \log_{10}(1000) = 6$, which is true—$A(1000) = 3.1415952...$ has 6 correct digits of $\pi$.

| $n$ | $A(n)$ | Error | $X(n)$ | $2\log n$ |
|---|---|---|---|---|
| 10 | 3.1696834530 | $2.81 \times 10^{-2}$ | 2 | 2 |
| 20 | 3.1481890536 | $6.60 \times 10^{-3}$ | 3 | 2.6021 |
| 50 | 3.1426296696 | $1.04 \times 10^{-3}$ | 3 | 3.3979 |
| 100 | 3.1418512561 | $2.59 \times 10^{-4}$ | 4 | 4 |
| 200 | 3.1416572635 | $6.46 \times 10^{-5}$ | 4 | 4.6021 |
| 500 | 3.1416029894 | $1.03 \times 10^{-5}$ | 4 | 5.3979 |
| 1000 | 3.1415952375 | $2.58 \times 10^{-6}$ | 6 | 6 |
| 2000 | 3.1415932996 | $6.46 \times 10^{-7}$ | 6 | 6.6021 |
| 5000 | 3.1415927569 | $1.03 \times 10^{-7}$ | 7 | 7.3979 |
| 10000 | 3.1415926794 | $2.58 \times 10^{-8}$ | 8 | 8 |

To prove that the three fractions are not equal to $\pi$, we checked consecutive values of $n$ until the appropriate polygon bound crossed each fraction. The tables shows the first value of $n$ for which the bound crosses the fraction, as well as the previous value. Since the sign changes between these two consecutive values, these are the smallest values of $n$ that prove each fraction is not equal to $\pi$.

| $n$ | 22/7 | $PC(n)$ | $PC(n) - 22/7$ |
|---|---|---|---|
| 90 | 3.1428571429 | 3.1428692543 | $1.21 \times 10^{-5}$ |
| 91 | 3.1428571429 | 3.1428413380 | $-1.58 \times 10^{-5}$ |

| $n$ | 333/106 | $PI(n)$ | $PI(n) - 333/106$ |
|---|---|---|---|
| 249 | 3.1415094340 | 3.1415093054 | $-1.29 \times 10^{-7}$ |
| 250 | 3.1415094340 | 3.1415099708 | $5.37 \times 10^{-7}$ |

| $n$ | 355/113 | $PC(n)$ | $PC(n) - 355/113$ |
|---|---|---|---|
| 6224 | 3.1415929204 | 3.1415929204 | $3.79 \times 10^{-11}$ |
| 6225 | 3.1415929204 | 3.1415929203 | $-4.78 \times 10^{-11}$ |

# 3 Using the Infinite Sum of Inverse Squares

The second method uses an infinite sum to approximate $\pi$. In 1644, Pietro Mengoli posed the following problem: Find the exact value of $\sum_{i=1}^{\infty} \frac{1}{i^2}$. This was called the *Basel Problem*, because Euler and the Bernoulli family, the people who worked on the problem, lived in Basel, Switzerland. In 1735, Euler solved it, showing that

$$\sum_{i=1}^{\infty} \frac{1}{i^2} = \frac{\pi^2}{6}.$$

When solving the Basel problem, Euler proved that

$$\sum_{i=1}^{\infty} \frac{1}{i^2} = \frac{\pi^2}{6}.$$

Let $S(n)$ be the partial sum of the first $n$ terms of this series, and let $L(n) = \sqrt{6S(n)}$. Since the series only grows as $n$ increases, $\pi > L(n)$ for all $n$.

For an upper bound, the remaining terms after $n$ (the "tail" of the sum) can be bounded using an integral comparison, since $1/x^2$ is positive and decreasing:

$$\sum_{i=n+1}^{\infty} \frac{1}{i^2} < \int_n^{\infty} \frac{1}{x^2}\, dx = \frac{1}{n}.$$

Adding this tail bound to $S(n)$ gives a value guaranteed to be larger than the true infinite sum:

$$U(n) = \sqrt{6\left(S(n) + \frac{1}{n}\right)}.$$

Therefore, $L(n) < \pi < U(n)$ for all integer $n$. To estimate $\pi$, we used the average of these two values:

$$B(n) = \frac{L(n) + U(n)}{2}.$$

We used Python to calculate the approximations and find the smallest value of $n$ needed to prove that each fraction is not equal to $\pi$. The table shows how the approximation improves as the number of terms increases. As $n$ increases, the approximation becomes more accurate and the error decreases.

Let $X(n)$ represent the number of digits of $\pi$ which are correct in $B(n)$. To approximate $X(n)$, we notice that $L(n) = \sqrt{6S(n)}$ and $S(n) = \pi^2/6 - T(n)$, where

$$T(n) = \sum_{i=n+1}^{\infty} \frac{1}{i^2} < \frac{1}{n}$$

is the tail of the series. Substituting, we get

$$L(n) = \sqrt{\pi^2 - 6T(n)} = \pi\sqrt{1 - \frac{6T(n)}{\pi^2}}.$$

Since $T(n)$ is small for large $n$, we can apply the approximation $\sqrt{1-x} \approx 1 - \dfrac{x}{2}$, giving

$$L(n) \approx \pi\left(1 - \frac{3T(n)}{\pi^2}\right) = \pi - \frac{3T(n)}{\pi}.$$

Because $T(n) \approx 1/n$ as mentioned above, the error is approximately $\pi - L(n) \approx 3/\pi n$. Taking $-\log$ of this error gives the number of correct digits:

$$X(n) \approx -\log\left(\frac{3}{\pi n}\right) = \log(n) + \log\left(\frac{\pi}{3}\right) \approx \log(n) + 0.02.$$

| $n$ | $B(n)$ | Error | $X(n)$ | $\log n + 0.02$ |
|---|---|---|---|---|
| 10 | 3.0977833565 | $4.38 \times 10^{-2}$ | 1 | 1.02 |
| 20 | 3.1187178681 | $2.29 \times 10^{-2}$ | 2 | 1.3210 |
| 50 | 3.1322044418 | $9.39 \times 10^{-3}$ | 2 | 1.7190 |
| 100 | 3.1368583862 | $4.73 \times 10^{-3}$ | 2 | 2.02 |
| 200 | 3.1392154383 | $2.38 \times 10^{-3}$ | 2 | 2.3210 |
| 500 | 3.1406393427 | $9.53 \times 10^{-4}$ | 3 | 2.7190 |
| 1000 | 3.1411155936 | $4.77 \times 10^{-4}$ | 4 | 3.02 |
| 2000 | 3.1413540224 | $2.39 \times 10^{-4}$ | 4 | 3.3210 |
| 5000 | 3.1414971768 | $9.55 \times 10^{-5}$ | 4 | 3.7190 |
| 10000 | 3.1415449112 | $4.77 \times 10^{-5}$ | 5 | 4.02 |

Therefore, summing $n$ terms of the inverse squares series gives roughly $\log_{10}(n) + 0.02$ correct decimal digits. This is much slower than the polygon method, which gained about $2\log_{10}(n)$ digits for the same $n$.

To prove that the three fractions are not equal to $\pi$, we checked consecutive values of $n$ until either $L(n)$ or $U(n)$ became greater than or less than each fraction, depending on whether the fraction was smaller or larger than $\pi$. The following tables show the first value for which the bound is above/below the fraction, along with the preceding value for comparison.

| $n$ | $22/7$ | $U(n)$ | $U(n) - 22/7$ |
|---|---|---|---|
| 19 | 3.1428571429 | 3.1428918112 | $3.47 \times 10^{-5}$ |
| 20 | 3.1428571429 | 3.1427662120 | $-9.09 \times 10^{-5}$ |

| $n$ | $333/106$ | $L(n)$ | $L(n) - 333/106$ |
|---|---|---|---|
| 11475 | 3.1415094340 | 3.1415094306 | $-3.37 \times 10^{-9}$ |
| 11476 | 3.1415094340 | 3.1415094378 | $3.88 \times 10^{-9}$ |

| $n$ | $355/113$ | $U(n)$ | $U(n) - 355/113$ |
|---|---|---|---|
| 1337 | 3.1415929204 | 3.1415929206 | $2.72 \times 10^{-10}$ |
| 1338 | 3.1415929204 | 3.1415929202 | $-1.27 \times 10^{-10}$ |

As mentioned above, $22/7$ and $355/113$ are disproven using $U(n)$, while $333/106$ is disproven using $L(n)$. Interestingly, proving $\pi \neq 333/106$ requires $n$ to be 11476, while proving $\pi \neq 355/113$ only requires $n$ to be 1338. This is in contrast to the inscribed-circumscribed method, where $n = 250$ was needed for $333/106$ and $n = 6225$ was needed for $355/113$. This suggests the lower bound $L(n)$ is less efficient than the upper bound, at least compared to the bounds using the inscribed-circumscribed method.

# 4 Using the Nilakantha Series

We now use the Nilakantha series to approximate $\pi$. This series was discovered by the Indian mathematician Nilakantha Somayaji around 1500 CE. It writes $\pi$

as an infinite sum of shrinking terms that alternate in sign:

$$\pi = 3 + \frac{4}{2 \cdot 3 \cdot 4} - \frac{4}{4 \cdot 5 \cdot 6} + \frac{4}{6 \cdot 7 \cdot 8} - \cdots$$

The partial sum after $n$ terms is

$$S(n) = 3 + \sum_{k=1}^{n} \frac{(-1)^{k+1} \cdot 4}{(2k)(2k+1)(2k+2)}.$$

Because the terms decrease in size and alternate in sign, the partial sums oscillate around $\pi$: the sum after an odd number of terms is always greater than $\pi$, and the sum after an even number of terms is always less than $\pi$. Therefore,

$$S_{\text{even}}(n) < \pi < S_{\text{odd}}(n).$$

These functions serve as our lower bound and upper bound for $\pi$. We used Python to calculate the approximations and find the smallest value of $n$ needed to prove that each fraction is not equal to $\pi$. The table shows how the Nilakantha approximation improves as the number of terms increases.

Let $X(n)$ represent the approximate number of correct decimal digits in $S(n)$. From the data, the approximations have about 4 correct decimal digits when $n = 10$, 7 when $n = 100$, 10 when $n = 1000$, and 13 when $n = 10000$. Each time $n$ is multiplied by 10, the number of correct digits increases by about 3. The data suggest the approximate function $X(n) \approx 3\log_{10}(n) + 1$. For example, when $n = 1000$,

$$X(1000) \approx 3\log_{10}(1000) = 3(3) + 1 = 10.$$

This converges much faster than the polygon method, which only gave about $2\log_{10}(n)$ correct digits.

| $n$ | $S(n)$ | Error | $X(n)$ | $3\log n + 1$ |
|---|---|---|---|---|
| 10 | 3.1414067185 | $1.86 \times 10^{-4}$ | 4 | 4 |
| 20 | 3.1415657347 | $2.69 \times 10^{-5}$ | 5 | 4.9031 |
| 50 | 3.1415907698 | $1.88 \times 10^{-6}$ | 6 | 6.0969 |
| 100 | 3.1415924110 | $2.43 \times 10^{-7}$ | 7 | 7 |
| 200 | 3.1415926228 | $3.08 \times 10^{-8}$ | 8 | 7.9031 |
| 500 | 3.1415926516 | $1.99 \times 10^{-9}$ | 9 | 9.0969 |
| 1000 | 3.1415926533 | $2.49 \times 10^{-10}$ | 10 | 10 |
| 2000 | 3.1415926536 | $3.12 \times 10^{-11}$ | 11 | 10.9031 |
| 5000 | 3.1415926536 | $2.00 \times 10^{-12}$ | 12 | 12.0969 |
| 10000 | 3.1415926536 | $2.50 \times 10^{-13}$ | 13 | 13 |

To prove that the three fractions are not equal to $\pi$, we checked consecutive values of $n$ (of the correct parity) until the appropriate bound crossed each fraction. Since 22/7 and 355/113 are both greater than $\pi$, the odd-term bound $S_{\text{odd}}(n)$ was used. Since 333/106 is less than $\pi$, the even-term bound $S_{\text{even}}(n)$ was used. The following tables show the first value for which the bound is above/below the fraction, along with the preceding value for comparison.

| $n$ | $22/7$ | $S(n)$ | $S(n) - 22/7$ |
|---|---|---|---|
| 3 | 3.1428571429 | 3.1452380952 | $2.38 \times 10^{-3}$ |
| 5 | 3.1428571429 | 3.1427128427 | $-1.44 \times 10^{-4}$ |

| $n$ | $333/106$ | $S(n)$ | $S(n) - 333/106$ |
|---|---|---|---|
| 12 | 3.1415094340 | 3.1414796890 | $-2.97 \times 10^{-5}$ |
| 14 | 3.1415094340 | 3.1415189856 | $9.55 \times 10^{-6}$ |

| $n$ | $355/113$ | $S(n)$ | $S(n) - 355/113$ |
|---|---|---|---|
| 95 | 3.1415929204 | 3.1415929361 | $1.58 \times 10^{-8}$ |
| 97 | 3.1415929204 | 3.1415929192 | $-1.18 \times 10^{-9}$ |

Again, since the series converges so quickly, the values of $n$ are drastically lower than they are for the other methods.

# 5 Using the Leibniz Formula

In the 15th century, Madhava of Sangamagrama discovered this series for $\pi$:

$$\pi = 4\left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots\right).$$

When it was rediscovered by Gottfried Wilhelm Leibniz in 1673, it was named after him.

Define the partial sum of this series as

$$L(n) = 4\sum_{k=0}^{n} \frac{(-1)^k}{2k+1}.$$

Since the series is alternating, even values of $n$ will serve as an upper bound for $\pi$, and odd values will be a lower bound. This sequence converges extremely slowly, much slower than the previous methods. The table shows the accuracy of the Leibniz series at various $n$:

Let $X(n)$ be the number of correct digits of $\pi$ in $L(n)$. From the data, we concluded that $X(n) \approx \log_{10} n$. For example, $X(1000) \approx \log_{10}(1000) = 3$. Since $B(1000) = 3.14112...$, which has 3 correct digits of $\pi$, this approximation is reasonably accurate.

| $n$ | $L(n)$ | Error | $X(n)$ | $\log n$ |
|---|---|---|---|---|
| 10 | 3.2323158094 | $9.07 \times 10^{-2}$ | 1 | 1 |
| 20 | 3.1891847823 | $4.76 \times 10^{-2}$ | 2 | 1.3010 |
| 50 | 3.1611986130 | $1.96 \times 10^{-2}$ | 2 | 1.6990 |
| 100 | 3.1514934011 | $9.90 \times 10^{-3}$ | 2 | 2 |
| 200 | 3.1465677472 | $4.98 \times 10^{-3}$ | 3 | 2.3010 |
| 500 | 3.1435886596 | $2.00 \times 10^{-3}$ | 3 | 2.6990 |
| 1000 | 3.1425916543 | $9.99 \times 10^{-4}$ | 3 | 3 |
| 2000 | 3.1420924037 | $5.00 \times 10^{-4}$ | 3 | 3.3010 |
| 5000 | 3.1417926136 | $2.00 \times 10^{-4}$ | 4 | 3.6990 |
| 10000 | 3.1416926436 | $1.00 \times 10^{-4}$ | 4 | 4 |

Finally, we proved that the three fractions were not equal to $\pi$. We did this by checking values of $n$ until the appropriate bound crossed each fraction. The tables show the first $n$ for which the bound eliminates the fraction, along with the preceding $n$ for comparison.

| $n$ | $22/7$ | $L(n)$ | $L(n) - 22/7$ |
|---|---|---|---|
| 788 | 3.1428571429 | 3.1428600802 | $2.94 \times 10^{-6}$ |
| 790 | 3.1428571429 | 3.1428568756 | $-2.67 \times 10^{-7}$ |

| $n$ | $333/106$ | $L(n)$ | $L(n) - 333/106$ |
|---|---|---|---|
| 12015 | 3.1415094340 | 3.1415094312 | $-2.74 \times 10^{-9}$ |
| 12017 | 3.1415094340 | 3.1415094451 | $1.11 \times 10^{-8}$ |

| $n$ | $355/113$ | $L(n)$ | $L(n) - 355/113$ |
|---|---|---|---|
| 3748626 | 3.1415929204 | 3.14159292035400 | $1.60 \times 10^{-14}$ |
| 3748628 | 3.1415929204 | 3.14159292035386 | $-1.27 \times 10^{-13}$ |

Since the series is so slow to converge, the required values of $n$ are much higher than the previous approximations.

# 6 Code

The following code was used for the inscribed and circumscribed polygons method:

```
import math

# Values of n used to generate the approximation table
values = [10,20,50,100,200,500,1000,2000,5000,10000]

print("n\tA(n)\t\t\tError")

for n in values:

    # Compute the inscribed and circumscribed approximations
    PI = n * math.sin(math.pi / n)
    PC = n * math.tan(math.pi / n)

    # Average of the two approximations
    A = (PI + PC) / 2

    # Difference between the approximation and pi
    error = abs(A - math.pi)

    print(f"{n}\t{A:.10f}\t{error:.2e}")
```

The following code was used for the inverse squares method:

```
import math

# Values of n used to generate the approximation table
values = [10,20,50,100,200,500,1000,2000,5000,10000]

def partial_sum(n):
    return sum(1 / (i * i) for i in range(1, n + 1))

print("n\tB(n)\t\t\tError")

for n in values:

    # Compute the lower and upper bracket bounds
    L = math.sqrt(6 * partial_sum(n))
    U = math.sqrt(6 * (partial_sum(n) + 1 / n))

    # Average of the two bounds
    B = (L + U) / 2

    # Difference between the approximation and pi
    error = abs(B - math.pi)

    print(f"{n}\t{B:.10f}\t{error:.2e}")
```

The following code was used for the Nilakantha method:

```
from mpmath import mp, mpf, fabs

mp.dps = 50

def nilakantha_partial(n_terms):
    total = mpf(3)
    sign = 1
    for k in range(1, n_terms + 1):
        term = mpf(4) / ((2 * k) * (2 * k + 1) * (2 * k + 2))
        total += sign * term
        sign *= -1
    return total

values = [10,20,50,100,200,500,1000,2000,5000,10000]

true_pi = mp.pi

print("n\tS(n)\t\t\tError")
for n in values:
    S = nilakantha_partial(n)
    error = fabs(S - true_pi)
```

```
    print(f"{n}\t{mp.nstr(S,-11)}\t{mp.nstr(error,-3)}")
```

The following code was used for the Leibniz method:

```
import java.util.*;
import java.math.*;

public class Main {
    public static void main(String[] args) {
      double sum = 0.0;
      int n = 100;
      double frac = 333.0 / 106.0
      for(int i = 0; i <= n; i++) {
        sum += Math.pow(-1,i) * 1.0/(2 * i + 1);
      }
      System.out.println(frac);
      System.out.println(4 * sum);
      System.out.println(4 * sum - frac);
    }
}
```

# 7 Conclusion

We have discussed the four methods we used to prove $\pi$ was not equal to various fractions. Below is a table containing all three fractions, the four methods, the least $n$ necessary, and the approximate function for $X(n)$. Three of these approximations are based on empirical evidence, but the approximation for the sum of inverse squares was proven to be accurate.

| | 22/7 | 333/106 | 355/113 | $X(n)$ |
|---|---|---|---|---|
| Inscribed-Circumscribed | 91 | 250 | 6225 | $2\log n$ |
| Inverse Squares | 20 | 11476 | 1338 | $\log n + 0.02$ |
| Nilakantha Series | 5 | 14 | 97 | $3\log n + 1$ |
| Leibniz Series | 790 | 12017 | 3748628 | $\log n$ |