Question: In Fibonacci sequence, it follows the rule that each number is equal to the sum of the preceding two numbers. Assuming the first two numbers are 0 and 1, what is the 50th number in Fibonacci sequence?

The first number is 0, the second number is 1, therefore, the third number is 0+1=1. The fourth number is 1+1=2. The fifth number is 1+2=3. The sixth number is 2+3=5. The seventh number is 3+5=8. The eighth number is 5+8=13.
….. (Skip 1000 tokens)
The 50th number is 32,432,268,459.

CoT

32,432,268,459

```
length_of_fibonacci_sequence = 50
fibonacci_sequence = np.zeros(length_of_)
fibonacci_sequence[0] = 0
fibonacci_sequence[1] = 1
For i in range(3, length_of_fibonacci_sequence):
    fibonacci_sequence[i] = fibonacci_sequence[i-1] +
    fibonacci_sequence[i-2]
ans = fibonacci_sequence[-1]
```

PoT

python

12,586,269,025

Question: Ketty saves 20000 dollars to the bank. After three years, the sum with compound interest rate is 1000 dollars more than the sum with simple interest rate. What is the interest rate of the bank?

Assuming the interest rate is x. The sum after two years with simple interest rate is 20000 + x * 20000 * 3 = 20000 + 60000x. The sum after two years with compoud interest rate is 20000 * (1 + x) ^ 3 = 200000 + 60000 * x + 60000x^2 + 20000x^3. The difference can be written as 60000x^2 + 20000x^3 = 1000. In order to solve x, we can use the quadratic formula. x = (-b +- sqrt(b^2 - 4ac)) / 2a, …. , x = (-20000 +- 6160) / 120000, x = -0.051333.

CoT

-0.051333

```
interest_rate = Symbol('x')
sum_in_two_years_with_simple_interest= 20000 +
interest_rate * 20000 * 3
sum_in_two_years_with_compound_interest = 20000 * (1 +
interest_rate)**3
# Since compound interest is 1000 more than simple interest.
ans = solve(sum_after_in_yeras_with_compound_interest -
sum_after_two_years_in_compound_interest - 1000,
interest_rate)
```

PoT

python SymPy

x = 0.24814