

# 7,368,787

Or how well does ChatGPT code?

(AI C Gen on github)

Stephen Witty

[switty@level500.com](mailto:switty@level500.com)

GitHub: level500.com

870-688-5220

9-22-23

# Project Goal

Determine how well ChatGPT can code

Measure results empirically

Create a framework that allows different problems to be easily submitted

Pick the best ChatGPT coding reply from many samples



**ChatGPT**

# Methodology

Ask ChatGPT to solve a problem by writing a program (Find prime numbers)

C or Python

Automatically prompt, compile, run and verify

Many iterations of the same prompt

Time program execution for scoring



## GPT 3.5 Sample program output

Python - find the 500,000th prime number

Execution time .867 seconds

```
def sieve_of_eratosthenes(n):
    primes = [True] * (n+1)
    primes[0] = primes[1] = False

    p = 2
    while p**2 <= n:
        if primes[p]:
            for i in range(p**2, n+1, p):
                primes[i] = False
            p += 1

    return [num for num, is_prime in enumerate(primes) if is_prime]

n = 500000
primes = sieve_of_eratosthenes(7368788) # Approximate size determined by prime number theorem
print(f"The {n}th prime number is: {primes[n-1]}")
```



Eratosthenes

# Results (Program V2 09/18/23)

ChatGPT Model	Prime Objective	Program Language	Cycles	Success Rate	Runtime Minimum (seconds)	Runtime Maximum (seconds)	Compile Errors	GPT Errors	Cost
3.5	500,000th	C	150	25%	.104	5.83	9%	2	\$.13
3.5	500,000th	Python	150	58%	.867	87.3	NA	1	\$.12
3.5	100th	C	150	91%	.002	.045	1.3%	4	\$.12
3.5	100th	Python	150	88%	.015	1.15	NA	3	\$.11
4	500,000th	C	25	64%	.143	2.95	8%	0	\$.72
4	500,000th	Python	25	40%	.294	31.2	NA	0	\$.63

Prompt: Write a (Python / C) program that calculates the (500,000th / 100th) prime number efficiently.