Prompt Design Experiment

Model type: GPT-4

Task: coding challenge – prime number check

1. Zero-shot prompt:

Prompt: write a python function to check if a number is prime.

AI response:

def is\_prime(n):

if n <= 1:

return False

for i in range(2, int(n\*\*0.5)+1):

if n % i == 0:

return False

return True

Observation: provides the correct function but lacks explanation or step by step breakdown.

1. Few-shot prompt:

Example 1:

Write a Python function to check if a number is even.

def is\_even(n):

return n % 2 == 0

Example 2:

Write a Python function to check if a number is odd.

def is\_odd(n):

return n % 2 != 0

Now write a Python function to check if a number is prime.

AI Response:

def is\_prime(n):

if n <= 1:

return False

for i in range(2, int(n\*\*0.5)+1):

if n % i == 0:

return False

return True

Observation:

AI follows the example style and delivers a concise, correct function.

1. Chain-of-thought prompt:

Prompt:

Let’s solve this step-by-step.

First, understand that a prime number is greater than 1 and only divisible by 1 and itself.

Then, we need to check for divisibility from 2 up to √n.

If any number divides n, it's not prime.

If no number divides it, it is prime.

Now write a Python function to check if a number is prime.

AI Response:

def is\_prime(n):

# Prime numbers are greater than 1

if n <= 1:

return False

# Check for factors from 2 to sqrt(n)

for i in range(2, int(n\*\*0.5) + 1):

if n % i == 0:

return False

return True

Observation:

This prompt yields a more thoughtful and explainable solution, ideal for understanding or teaching.

Overall analysis:

* Zero-shot prompting: effective for generating quick but lacks explanation.
* Few-shot prompting: AI follows the example style and delivers a concise, correct function.
* Chain-of-thought prompting: the most detailed and structured response, perfect for complex problems.