Prompt structure

Instruction for a model

You are a logician with a background in mathematics that translates natural language reasoning text to Python3 code so that these natural language reasoning problems can be solved.
During the translation, please pay close attention to defining variables and rules.
Do not add any comments from you.
Be guided by the following example:
Example input text:

Anonymized In-context examples

Input premises

Input premises for Task1

Input text:
Textual context: It is unlikely for a vehicle to operate for less than 2 hours or more than 10 hours in a typical day.
A vehicle idling for less than 1 hour or more than 5 hours in a day is not common.
Each typical day, a vehicle stops between 20 and 300 times.
On a typical day, the load on axle 1 ranges from 2000 to 8000 kg.

Input premises for Task2

Input text:
Textual context: Axle 1's load cannot exceed Axle 2's load.
The payload must always be smaller than or equal to the load on axle 2.
The total operation time must always be not lower than the combined duration of PTO operation, idle time, and driving time.

Anonymized In-context examples for Task1

```python
# A vehicle operates over a A param between 20 and 110 in a typical day.
def r1(a: float) -> bool:
    return 20 <= a <= 110


# A vehicle operates for a B param between 2 and 10 in a typical day.
def r2(b: float) -> bool:
    return 2 <= b <= 10


# A vehicle operates with C param between 0.2 and 5 in a typical day.
def r3(c: float) -> bool:
    return 0.2 <= c <= 5


# A vehicle remains D param between 1 and 5 in a typical day.
def r4(d: float) -> bool:
    return 1 <= d <= 5
```

Anonymized In-context examples for Task2

```python
# Parameter E's value cannot be less than parameter F's value.
def r1(e: float, f: float) -> bool:
    if e >= f:
        return True
    else:
        return False


# Parameter M's value cannot be greater than parameter N's value.
def r2(m: float, n: float) -> bool:
    if m <= n:
        return True
    else:
        return False


# Parameter W cannot be less than the sum of parameter X, parameter Y and parameter Z.
def r3(w: float, x: float, y: float, z: float) -> bool:
    if w >= x + y + z:
        return True
    else:
        return False
```