Arithmetic Evaluation

Key Points:

- The model's behavior changes based on the available functions.

- When computing an expression, the model will use the most appropriate tool for each operation if that tool is available.

Example: Given the expression **4 ** 4 - 6 * 2 + 8**

- If the available tools are add_tool, multiply_tool, divide_tool, and subtract_tool, the model would use the multiply_tool for the multiplication function to compute **4 ** 4**

- If the available tools include add_tool, multiply_tool, divide_tool, subtract_tool, and exponential_tool, the model would use the power function to compute **4 ** 4**

Mathematical Reasoning

Key Points:

- The model is not good at zero shot prompting when it has to compute complex expressions

- Possible solutions:
    + Provide a good api call to handle the edge case and avoid multiple function calls from the agent
    + Provide chain of thoughts / few shot prompting

Example of Limited Tokens and Simple Calculation:

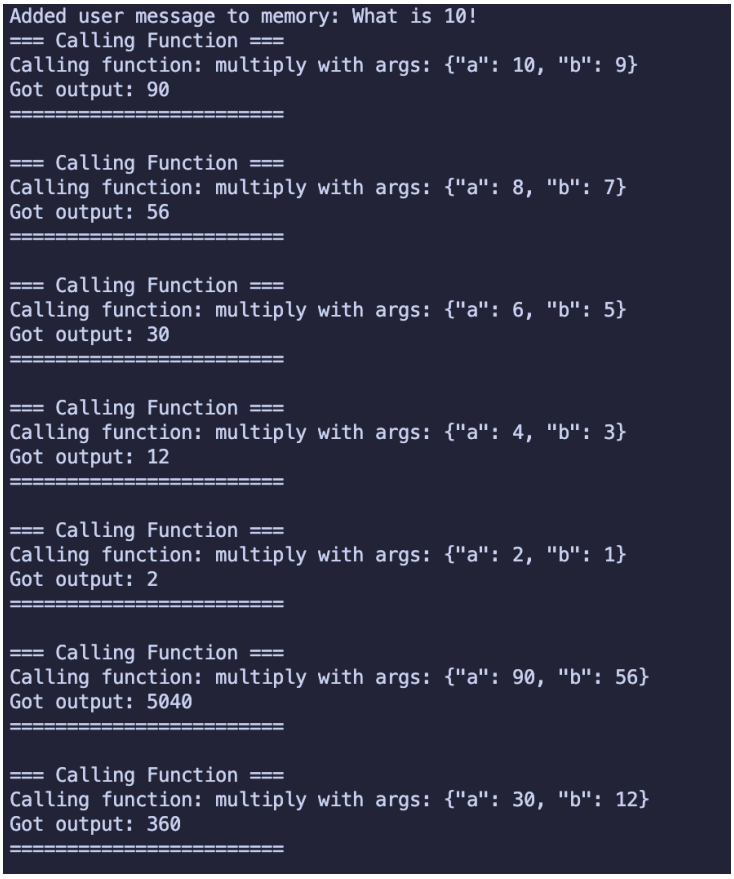

Figure 1: Compute the factorial of a number using only the multiply function. However, the computation halts at 30 * 12 = 360 due to token limitations. The available tools include add, subtract, divide, and multiply.

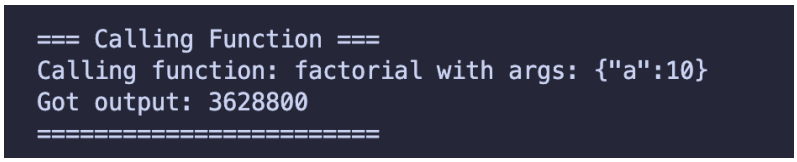

Figure 2: Compute the factorial of a number using a single function call to the factorial function. The available tools include add, subtract, divide, multiply, and factorial.

Current Problems Need To Look At: define an input parser (custom tool calls and llm pydantic), improve the reasoning