I have choose this ([WA3XC.3](#)) WA problem for GPT 4 to work on directly. (This is the last problem in the WA, so it might be harder for the AI to work on that directly without doing previous questions first).

The conversation is recorded here. [https://chat.openai.com/share/afa21ebe-a6e7-4d9f-a2cc-87118a8f ef6c](https://chat.openai.com/share/afa21ebe-a6e7-4d9f-a2cc-87118a8fef6c). In the end, I tried to get the GPT 4 to the right answer. (with some tiny mistakes. GPT 4 will tell you which rule it used in each step, sometimes it will generate correct expression but it will tell the wrong rule that reaches this expression).

At first, I gives all the rules defined, and an initial expression to work on. I give the GPT 4 the context that it's a OCaml like program. GPT 4 seems doing well at first. The first mistake it makes is doing

```
Eval(m (Eval(y, {m -> <x -> fun y -> x * y, {x -> 3, y -> 5}>, x -> 3, y -> 5}))
x, {m -> <x -> fun y -> x * y, {x -> 3, y -> 5}>, x -> 3, y -> 5})
```

which it decides to evaluate the middle `y` expression first. This is easily solved by asking it to do the application on x first.

Another mistake it makes is that:

it tries to go from

```
Eval(Eval(m (Val 5), {m -> <x -> fun y -> x * y, {x -> 3, y -> 5}>, x -> 3, y ->
5}) (Val 3), {m -> <x -> fun y -> x * y, {x -> 3, y -> 5}>, x -> 3, y -> 5})
```

directly to

```
Eval(Eval(m, {m -> <x -> fun y -> x * y, {x -> 3, y -> 5}>, x -> 3, y -> 5}) (Val
5) (Val 3), {m -> <x -> fun y -> x * y, {x -> 3, y -> 5}>, x -> 3, y -> 5})
```

It seems to have a hard time understanding that how to evaluate `m`. At first I tried to hint the GPT 4 the rule it should actually use, but this only confuses the AI. In the end, I explicitly tells it that `Eval(Eval(Eval m, env) Val 5, env), Val 3, env))` is the correct form, which finally works.

Another type of mistakes it will have is

```
Eval(fun y -> 5 * y, {m -> <x -> fun y -> x * y, {x -> 3, y -> 5}>, x -> 3, y ->
5}) (Val 3)
```

It tries to substitute the value into the expression directly rather than put the value into the closure / env. This also takes several tries, and in the end, I have to give the right form explicitly.

Two thing I notice is that, though I mentioned it is OCaml-like program, the GPT 4 has a hard time to understand 2 high-level concepts:

- Evaluation Order: In OCaml, typically evaluate from right to left, then apply from left to right.
- Closure / Env: It tries to substitute the number into the expression directly rather than putting it into the closure and env.

Another interesting trend is that, the GPT 4 will become less reflective on your correction / hints / help as the conversation goes. This might due to the reason that as conversation goes on, its internal "knowledge memory" is unable to hold all the information, and there are too much noise in the conversation, so it will have less reasoning ability.