Meno II: A self-referential Socratic dialogue about memory and computer programming

Anonymous

March 2025

Abstract

We provide a Socratic dialogue in which Meno challenges Socrates' principle that all learning is actually the remembering of things known but forgotten. Meno claims if Socrates' principle holds, then Meno should already know the contents of the ongoing dialogue. Meno challenges Socrates to help Meno recall those contents, claiming Socrates cannot do this, lest Meno could immediately say the opposite of whatever he supposedly recalls he was about to say. Socrates eventually succeeds at Meno's challenge in an unexpected manner.

1 Introduction

In the Meno, Plato's Socrates suggests the principle of anamnesis: there is no learning, only the remembering of forgotten knowledge. Presumably this should not be understood of contingent facts, or rather, perhaps we never truly know contingent facts: thus, Wittgenstein refused to acknowledge that there is no rhinoceros in the room. But it is fun to speculate about what would be implied if anamnesis did extend to contingent facts. It would extend to Plato's speakers' knowledge of their own words: they would know (but have forgotten) the very words they say, before saying them.

We designed a Socratic dialogue in which Meno asks for assistance remembering what he will say in the dialogue. Against all odds, Socrates succeeds. The dialogue takes place in a universe where Socrates and Meno know computer programming; Socrates uses techniques similar to quines (self-printing computer programs) and obfuscated code to evade Meno's paradox.

2 Dialogue

Meno: Good morning, Socrates.

Socrates: Good morning to you, Meno, my friend.

Meno: I've been wanting to talk to you.

Socrates: And just what would you like to talk about, my dear friend?

Meno: It's about what you said the last time you and I spoke together.
Socrates: I suppose you mean when I told you that learning is really just remembering?
Meno: Yes, and that we actually know everything already, and we merely remember it.
Socrates: And I suppose you've contrived some sort of paradox or something to try and disprove it?
Meno: Quite. Once we finish this delightful discussion we're having, don't you think that then I will know all the things we said?
Socrates: You mean, know the conversation?
Meno: Exactly. Once we're done, I'll then know everything we said. That is, I will have learned it, wouldn't you say?
Socrates: I'm not really so sure about that. More often than not, I find I misremember conversations, or, even worse, I think that I remember and understand them, and then it later happens that I either remember wrong, or understand wrong. Besides, your question makes me uneasy, for I feel as if I am in danger of falling into a trap somehow.
Meno: What! Does the great Socrates have trouble with his memory?
Socrates: I often wish that I could remember things better, Meno. For I feel sometimes as though I flitter between two different worlds. One is intelligible and beautiful, the other is sensuous and profane. For instance, when my house is well organized, and I remember where I have put everything, it's as if I live in a house designed by the geometers, for I can find my shirt or my shoes by an act of pure thinking. But more often, it pains me to say, my house is not organized, and I must grope around for my shirt or my shoes, and I feel as if I'm cast out of that house of the geometers, and plunged into a carnal house of the senses.
Meno: Well, as for me, I take great pride in my house always being organized. For I employ servants who keep it organized for me. And, look here: I have a servant writing this very conversation of ours on a stone tablet. You know him: it is the same boy with whom you conversed about geometry the last time you visited. I treasure your words like gold, Socrates, and I will preserve them, and guard them, just like I would treat a valuable treasure.
Socrates: You flatter me, Meno. But I suspect you're just saying that because you want to catch me in some trap. Very well, Meno. For the sake of the argument, I grant that after we have this conversation, you shall have learned it.
Meno: Well then, doesn't that mean I already knew it, and have only to remember?
Socrates: Yes, I grant that, Meno. Even before you were born, you already knew I would say the words, 'Yes, I grant that, Meno,' which I said just now.
Meno: Well then, dear Socrates, I have a challenge for you. With your famous method of questioning, please guide me to recall all of the things we shall say in this dialogue we're having. That is, if you are able to. But I doubt it.
Socrates: Why do you doubt I can do it, my dear Meno?

Meno: Because the very moment you help me remember the things we're going to say to one another, I will immediately remember what line I will say next, and I will say the very opposite.

Socrates: That's very clever, Meno. Let me see if I understand right. So if I help you recall what things we're about to say to each other, and thus you remember that in response to my thus helping you, you are next going to say 'Thank you Socrates,' then, knowing that, you will deliberately say, 'No thank you, Socrates'?

Meno: Exactly. Or suppose that you help me remember all the things we're going to say, and, in particular, I then remember that as soon as you finish thus helping me, I will then say 'You win, Socrates!' And then, rather than saying 'You win, Socrates!', I will say, 'You lose, Socrates!' Nor will I be lying, for my very saying 'You lose, Socrates!' will prove that you did indeed lose, because if the memory you made me remember was accurate, I would have instead said 'You win.'

Socrates: But what if I help you recall all the things we're going to say, and, as a part of that, in particular, you remember that as soon as I finish thus helping you, you will say, 'You lose, Socrates!' Then, according to your plan, you will say the opposite, which is, 'You win, Socrates!' Wouldn't I then have won, by your own admission, Meno?

Meno: No, Socrates. The purpose of this challenge is not for you to make me say the phrase, "You win, Socrates!" Rather, the purpose of this challenge is for you to guide me toward remembering what I'm going to say, whether that's "You lose, Socrates!" or "You win, Socrates!", it doesn't matter. And when I deliberately say the precise opposite of what I remember I was going to say next, then you will lose the challenge, my friend, whether that opposite statement is "You win, Socrates!" or "You lose, Socrates!"

Socrates: Very well. This is a fun challenge. And I think if I'm to have any hope at succeeding, I will need to resort to computer programming.

Meno: Perhaps. For my challenge does seem somehow related to things like the recursion theorem and the incompleteness theorem. But even computer programming won't rescue you from my paradox.

Socrates: We'll see. But if I do succeed at this task, it would be quite impressive, wouldn't it?

Meno: Quite.

Socrates: And the reason it would be impressive is because I helped you predict the very future, right?

Meno: Yes.

Socrates: And would it be equally impressive if I helped you to remember the past? For instance, if I helped you recall that you began this whole conversation by saying 'Good morning, Socrates'?

Meno: No.

Socrates: Well then, by Olympus, wouldn't it at least be impressive if I helped you remember a large amount of the past? For example, if I helped you to remember this entire conversation from 'Good morning, Socrates,' all the way up to 'Well then, by Olympus'? Would that not be impressive?

Meno: No.
Socrates: So what you've said in the past is, in some sense, trivial?
Meno: Yes.
Socrates: But what you'll say in the future is deep and mysterious?
Meno: Right.
Socrates: And if I tried to invoke something deep and mysterious, then that would be cheating, right? For instance, were I to say, 'You will next say the thing that you will next say,' that would certainly not be impressive, would it?
Meno: That would be most unimpressive.
Socrates: But if I invoked something trivial, that would be fine, right? For example, if, through my questioning, I remind you that we are standing in Athens, that would be perfectly reasonable, right?
Meno: Perfectly reasonable.
Socrates: So for example, suppose I ask you: take the first sentence you said to me, and repeat it to me again right now, and then after that, please repeat it to me a second time. And suppose I claimed that, by doing so, you would recite our entire conversation. Would I be right?
Meno: Absolutely not.
Socrates: Well, why not?
Meno: Well, suppose I did exactly what you propose. Then I'd say: 'Good morning, Socrates. Good morning, Socrates.'
Socrates: Well, wouldn't that be a conversation? Albeit, a little one sided?
Meno: It would be a very short, one sided conversation.
Socrates: But I'd be wrong to claim that that brief conversation was our entire conversation we are having?
Meno: Absolutely wrong.
Socrates: Why?
Meno: Well, for one thing, at the start of this conversation we're having, I did not greet you twice in a row like that. For another thing, you replied to me, but in the brief conversation in question, you did not respond.
Socrates: So the reason why this supposed one sided conversation would not win your challenge, is because of the content of the results?
Meno: Quite so.
Socrates: And is there any other reason why this example conversation would not beat your challenge?
Meno: No.
Socrates: In that one sided example, I used earlier pieces of our conversation as tools. But you say, the lone reason that example would fail to solve the challenge, is due to the contents of the resulting one sided conversation?
Meno: That's right.
Socrates: Then it's perfectly alright for me to use earlier pieces of our conversation like that?
Meno: Yes, that's alright.
Socrates: Well, supposing I said to you: 'Take our whole conversation up until I said the word octopus. Repeat that whole conversation back to me. Then

repeat that whole conversation back to me a second time.' Would that be a solution to your challenge?
Meno: Decidedly not.
Socrates: Because the content of the resulting conversation would not equal the actual conversation?
Meno: Yes. For example, in the resulting conversation, you would say: '...up until I said the word octopus.' And then, since at that point I would have finished the first repetition and it would be time for the second one, I would say: 'Good morning, Socrates.'
Socrates: Whereas, in the actual conversation, after I said "octopus", you did not immediately respond 'Good morning, Socrates'?
Meno: That's right.
Socrates: Then we shall have to think of a more clever method. But first, let's consider the example more deeply. Let X denote our whole conversation up until, and including, when I said "octopus". And suppose I wanted to ask you to repeat X to me, and then to repeat X to me a second time. Can you program a function that would take X and repeat it that way?
Meno: Certainly, that's easy. How about the following?

```
def repeat_twice(X):
print(X)
print(X)
```

Socrates: So then, if I asked you to take the first sentence you said to me, and plug it in to this 'repeat_twice' function, would the output be as follows?

```
Good morning, Socrates.
Good morning, Socrates.
```

Meno: Quite so.
Socrates: What if I asked you to take the nineteenth sentence you said to me, and plug that into 'repeat_twice'? What would be the result?
Meno: I'm afraid I don't remember exactly what that nineteenth sentence was.
Socrates: Can you think of any way of recalling it?
Meno: Well, yes. We could read it from my servant boy's tablet here. He's been diligently writing down this conversation for me. You know I value your words more than gold, Socrates.
Socrates: I should warn you not to read or write too much philosophy, Meno. For you cannot question a book, neither can the book question you. But in this case I admit it's convenient. You say by consulting your servant's tablet, we can recall exactly what the nineteenth sentence you said was?
Meno: Yes.
Socrates: But isn't your servant writing some other things here? Look. He writes 'Meno:' whenever you begin talking. He writes 'Socrates:' whenever I do.

Meno: Sure.

Socrates: We have a serious problem, then. Look at the tablet right here, near the very top. It says: 'Good morning to you, Meno, my friend. Meno:' Now, I can't remember. Did I say 'Meno:' immediately after 'Good morning to you, Meno, my friend'? Or rather, does 'Meno:' here mean that you began talking?

Meno: I believe it means that I started talking. I don't think you literally said 'Meno, my friend, Meno'.

Socrates: But look here, near the bottom of your tablet. It says, quote: Did I say 'Meno:' immediately after 'Good morning to you, Meno, my friend'? End quote. Well, then. I know it was me who said 'Did I say', but who said 'immediately'? Did Socrates say, 'Did I say Meno: immediately after...'? Or did I merely say 'Did I say' and then Meno interrupted and said 'immediately after'?

Meno: I'm quite sure you said 'Did I say Meno: immediately after...'

Socrates: But the tablet itself does not tell us for certain?

Meno: I guess it might not.

Socrates: And just think. If you were reading this tablet by your lonesome, you couldn't even ask it whether this 'Socrates:' here means that Meno said 'Socrates:', or whether it means Socrates started talking. Now do you see the danger, then, of reading too much philosophy?

Meno: Perhaps.

Socrates: And your challenge is quite hopeless. For if I claim to remind you what our whole conversation is, you won't be able to check whether my claim holds true. Because you can't tell from this tablet who said what when, can you?

Meno: I guess not.

Socrates: But maybe we can modify the challenge. Do you at least know what is written on the tablet? For instance, that the very first letter on the tablet is 'M'?

Meno: Indeed.

Socrates: Then maybe instead of helping you to recall our future conversation, I could instead help you to recall the future contents of the tablet. Your little paradox would still confound me, wouldn't it?

Meno: It would.

Socrates: And I wouldn't even necessarily need to help you remember my own contributions to the tablet, would I? Wouldn't you say I won if I could just help you recall your own future contributions to the tablet?

Meno: I'm not entirely sure what you mean by that, Socrates.

Socrates: Let me give you an example. Suppose I helped you to remember that your own future contributions to the tablet would be this:

```
Wow!
You win, Socrates!
```

And suppose that immediately after I so helped you, you were to say, "Wow!" And next, suppose that I said, "Do I win, Meno?" And then you said, "You win,

Socrates!" In that scenario, would you grant me victory? Even though I did not help you remember my future contributions, namely "Do I win, Meno?", but only your own contributions?
Meno: I suppose I would not begrudge you the victory, Socrates.
Socrates: Then suppose that I were to announce: "Meno's next contribution to the tablet shall be 'You win, Socrates.' Meno: You win, Socrates." That is to say, first, I announce that Meno's next contribution on the tablet will be, "You win, Socrates". And at that point, I suppose you would want to pipe in and say "You lose, Socrates". But before you have a chance to do that, I quickly say, "Meno: You win, Socrates". Would I not win then?
Meno: Hardly! That would be cheating, Socrates. You know they say, only a fool puts words in other people's mouths. We would both know that you had cheated and had not won. And before you say that someone reading this tablet here would think you'd won, for they were confused by your syntactical trick, the truth is, not even a reader of the tablet after the fact would think that you won. And that's because your challenge is not to help me remember some small part of my future contributions to the tablet. No, your challenge is to help me remember the entirety of my future contributions to the tablet, and in the right order, too. So in that example of yours, after you say "You win, Socrates. Meno: You win, Socrates," I would then say, "You cheat, Socrates." Then even someone reading the tablet years later, who was not here in person, could see you did not win. Because they would see that I contributed the words, "You cheat, Socrates" to the tablet, and you did not predict those.
Socrates: You are cunning, Meno. I doubt even Ulysses could outwit you.
Meno: Why do you say that, Socrates?
Socrates: Do you remember how Ulysses tricked Polyphemus, the cyclops? He lied to Polyphemus, saying his name was "Noman". Then when Polyphemus's neighbors came and asked if some man was slaying him, Polyphemus said, 'Noman is slaying me.' Didn't Ulysses do that on purpose, so that the cyclops's friends would misunderstand, and think Polyphemus had said "No man is slaying me"? Isn't that just the sort of syntactical semantical confusion you have so effortlessly avoided just now?
Meno: I reckon it is.
Socrates: I think this challenge is very difficult, and I almost want to give up, except a certain voice in my head compels me not to do that. Certainly this task is beyond my own power, and I can only hope that it is within your power, Meno. So I will ask you questions and we will find out whether you have the power to figure out how to defeat yourself in this challenge, though I daresay you have a conflict of interest in the matter. Let us start from the beginning. Doesn't the first line on the tablet say, "Meno: Good morning, Socrates"?
Meno: We've already established that.
Socrates: If I requested you plug that line into your 'repeat_twice' function, would the result be as follows?

```
Meno:  Good morning, Socrates.
Meno:  Good morning, Socrates.
```

Meno: Yes, I believe so.
Socrates: What if instead of you, a cobbler were to plug that line into your 'repeat_twice'? Would the result still be the following?

```
Meno:  Good morning, Socrates.
Meno:  Good morning, Socrates.
```

Meno: Yes, of course.
Socrates: And then what if a painter plugged that line into 'repeat_twice'? Would the result then be the following?

```
Meno:  Good morning, Socrates.
Meno:  Good morning, Socrates.
```

Meno: Yes, the output would still not change, even if we were painters.
Socrates: And what if a sailor plugged that line into 'repeat_twice'? Would the result be the following?

```
Meno:  Good morning, Socrates.
```

Meno: Good morning, Socrates.
Socrates: Hey! Don't you know it's impolite to finish someone's sentence for him like that?
Meno: Forgive me, Socrates. But you were being rather predictable.
Socrates: I was doing that on purpose, because I suspected you'd react that way. You see, I wanted to show you something. Look on your tablet near the end here, where you finished my sentence for me just now. Don't you think someone could get confused, reading this?
Meno: Quite confused.
Socrates: So then, reading philosophy from a tablet, without being able to ask the philosopher questions, is not much good, is it?
Meno: Sure, Socrates, if we assume the tablet was deliberately designed to be confusing! You know, I'm almost starting to understand why they accuse you of corrupting the youth.
Socrates: Meno, that's a ridiculous rumour. All I have ever done to the youth is encourage them to think a little. And anyone can think whatever they want, and then turn around and unthink it just as easily.
Meno: Certainly.
Socrates: And all these things we've been saying are just idle thoughts, aren't they? You could ponder these things deeply all day and all night, and then the very next morning, you could ponder the exact opposite.
Meno: You're right.

Socrates: So one cannot corrupt the youth, then, just by contemplating thoughts together with them. For one can change one's thoughts just as effortlessly as one can imagine a mountain made of gold, wouldn't you agree?
Meno: If you put things like that, yes.
Socrates: And changing people's thoughts is like writing Greek on the surface of the water. The water is so pliable, you can write on it with your bare finger. But the instant that you've written there, your writing disappears, doesn't it?
Meno: It does.
Socrates: In fact, I always assumed that's what Thales meant, when he said that everything is made out of water. He meant that all the impressions we make on the world fade, just as if we inscribed them on the face of the sea. You could not corrupt the Aegean by writing on its surface, no matter how scandalous the things you wrote there. Why, even the language we speak slowly changes. And we try to nail that down, by imposing sharp constraints on it in the form of poetry. But over long enough time, even that fails, and it's like trying to keep the tide from going out by nailing the water to the sand. In the same way, I believe you could never corrupt a man just by writing thoughts into his mind, Meno.
Meno: But surely you do admit some corrupt youths exist, don't you, Socrates?
Socrates: I admit it.
Meno: Well, how could a youth be corrupted, if not through his mind?
Socrates: Youths have bodies, too, besides just their minds, don't they?
Meno: Of course.
Socrates: So to really corrupt the youth, you'd have to corrupt their bodies, not their minds.
Meno: Perhaps.
Socrates: And isn't the eye by far the most fragile and sensitive part of the body?
Meno: So it would appear.
Socrates: And isn't the slightest damage to the eye more grievous than a terrible wound elsewhere on your body?
Meno: Within reasonable limit, sure.
Socrates: So then if someone were going to corrupt the youth, they should try to corrupt the eyes of the youth, shouldn't they?
Meno: Maybe.
Socrates: And maybe that's why people think I have such a corrupting influence, not because of the words I say, but because of these bulging, bugging eyes I've got. And because my eyes bulge out, people assume their childrens' eyes will bulge out from talking to me. As if it isn't we who converse, but as if our eyes are men conversing together. But no man could convince a person to deliberately harm his own eye, could he?
Meno: No man except maybe Noman.
Socrates: Then, to corrupt a man's eye, one would have to trick him into doing something that hurts his eye without him knowing it?
Meno: I reckon so, Socrates.

Socrates: One would have to persuade him to do something that damages his eye slowly. One would have to persuade him to damage his eye in some way such that he didn't even realize he was doing so. Otherwise, the victim would quickly turn against you, wouldn't he?
Meno: Yes, and he would despise you, too.
Socrates: Can you think of any way a man could damage his own eye like that?
Meno: He could stare at the sun, Socrates, I suppose that's what you're trying to hint at.
Socrates: So then, to corrupt the youth in the most efficient way, one should persuade them to stare at the sun?
Meno: Yes. That is, if one wants to corrupt the youth, Socrates. Not that either of us wants that.
Socrates: But if you convinced a young man to stare at the sun, wouldn't wiser men come to his rescue? Wouldn't they tell him, 'Stop! You must not stare at the sun! It will blind you!'?
Meno: Indeed.
Socrates: And so in order that your diabolical plan should work, you would have to anticipate that, wouldn't you? Much like we're trying to anticipate Meno here. You'd have to not just persuade your young victim he should gaze at the sun, but even predict his elders will tell him 'Stop!' You'd have to make him think his elders were wrong, wouldn't you?
Meno: There would be no other way to do it.
Socrates: But he loves his elders. He'd never believe they would intentionally lie to him. You'd have to convince him his elders meant well, but that they themselves had been deceived.
Meno: I suppose you'd have to.
Socrates: You'd have to convince him that they're like men in a cave, staring at shadows on the wall. That the sun is a fire behind them, casting those shadows. That their eyes have grown accustomed to the dark, and that in order to escape the cave, they must turn and face the sun, wouldn't you?
Meno: That is one way to do it.
Socrates: And when their elders cried, 'Stop!', they would say to themselves: 'These elders don't realize we're in a cave. They believe this fire is the sun, and it will burn up my pupils.' Wouldn't they?
Meno: That would be our stratagem, yes.
Socrates: And if some youth ever returned to the cave, after dwelling on the bright surface, they really would be blind, while their eyes adjusted to the darkness, wouldn't they?
Meno: Quite blind, yes.
Socrates: And their elders would say, 'You should have listened to us! We warned you you would go blind if you stared at the sun like that! Oh, if only you had listened!'
Meno: Yes, and their poor elders would be quite upset.
Socrates: What would happen to whoever told that youth to turn and face the fire in the cave? He'd be accused of corrupting the youth, would he not?

Meno: At the very minimum.
Socrates: Because the youth's eyes were damaged? Or because of the thoughts in the youth's mind?
Meno: Because the youth's eyes were damaged, clearly.
Socrates: So then, there's very little risk I'll corrupt you, Meno, merely by making you think about lines of text on a tablet, is there?
Meno: None at all.
Socrates: Then let's proceed. For, the last thing I would ever want to do to you is corrupt you, Meno.
Meno: Yes, Socrates, let's get back to the issue at hand. Don't dodge my challenge by pursuing some wild tangent. Let me remind you of your precarious position. You claimed, last time we met, that there is no such thing as learning, rather, there is only remembering. I have pointed out that by that logic, I already know all the contents of this conversation we're having. And I have challenged you to help me remember those contents. And now you're stuck, for should you succeed in this task, I shall immediately say the opposite of whatever you help me remember I was about to say next.
Socrates: Thank you, dear Meno, for holding me accountable.
Meno: Let's stop tarrying, then!
Socrates: Yes, we must proceed. Well, then, suppose rather than handing you the first line of this conversation as X, suppose I handed you the whole conversation as X, up until when I said 'Yes, we must proceed'. And suppose I wanted your function to output the first five characters of X. What then would it output?
Meno: It would output 'Meno:'
Socrates: And what if I wanted it to output the first ten characters?
Meno: It would output 'Meno:Goodm'.
Socrates: And what would that function look like, that outputs those first ten characters?
Meno: It would be this:

```
def output_first_ten(X):
print(X[:10])
```

Socrates: But look here. There's some space on your servant's tablet between 'Meno:' and 'Good'. And if someone included that space in the X they passed to you, then your function would not output 'Meno:Goodm', but rather it would output 'Meno: Good', wouldn't it?
Meno: Yes.
Socrates: Can you modify your function to discard all such spaces? We'd better discard tabs and linebreaks too, and also hyphens, since I see your servant sometimes inserts a hyphen when a line ends in the middle of a word.
Meno: Sure. Here's the modified version:

```
def output_first_ten_without_space(X):
X=X.replace(" ","") # Remove spaces
```

```
X=X.replace("\n","") # Remove linebreaks
X=X.replace("\t","") # Remove tabs
X=X.replace("-","") # Remove hyphens
print(X[:10])
```

Socrates: And if we changed ten to forty, what would your function output then?
Meno: It would output "Meno:Goodmorning,Socrates.Socrates:Goodm".
Socrates: And it's good that we removed the linebreaks, for otherwise I suppose it would have output the following:

```
Meno:Goodmorning,Socrates.
Socrates:Good
```

And I think that would be much tougher to wrap my head around. I think we're making some progress. We are not limited to printing fragments of earlier conversation directly, are we? We could alter those fragments first, before printing them, right? For instance, you could write a program that prints the first ten characters of the discussion, but transformed to uppercase. What would the result be?
Meno: It would be 'MENO:GOODM'.
Socrates: In that case, could we say that you have taken an initial fragment of our conversation, and you have transformed it in a certain way, and that 'MENO:GOODM' is the output?
Meno: Yes.
Socrates: Now as to your challenge, suppose I described a more elaborate transformation. And let's suppose you performed that transformation on this conversation so far, from the start until the moment when I said "Now as to your challenge". And suppose that the output was:

```
You win, Socrates.
You're welcome.
I admit it.
```

Now suppose that once you performed that transformation and realized the output, you immediately said, "You win, Socrates." And suppose after that, I said, "Thank you, Meno." And suppose after that, you said, "You're welcome." And suppose after that, I said, "Do you admit that there is no learning, but only remembering?" Suppose after that, you answered, "I admit it." Then I would have passed your challenge?
Meno: Yes, but that would never happen.
Socrates: Why not?
Meno: In that situation, after seeing that output, which predicts I will say "You win, Socrates", I'd respond by saying the opposite. I'd say, "You lose, Socrates". And if instead the output started with "You lose, Socrates", then

I would say the opposite of that, and say, "You win, Socrates", but we'd both know you really lost.
Socrates: Yes, that's what makes this challenge so tricky, Meno.
Meno: Are you ready to give up, Socrates?
Socrates: It is not I who has the choice to give up or to continue, Meno, it is you. For I am only asking questions, you are the one answering. Indeed, I think that's why this approach cannot work. For you are not a book, Meno. You are not just some lines of text. You do not choose your words blindly, like someone blind from sun gazing; rather, you calculate them in response to things. Before you speak, you consider in your mind what you are going to say, mentally performing calculations that are invisible to me. And in the context of computer programming, what do we call something that acts that way, doing scratchwork internally to compute what to do next?
Meno: We call that a program, Socrates.
Socrates: Well, let's change our approach then. Suppose we described some elaborate transformation. And suppose you performed that transformation on this conversation so far, from the start until when I said, "Well, let us change our approach then." And suppose the output of that was:

```
print("""You win, Socrates.
You're welcome.
I admit it.""")
```

And suppose that the moment you knew that output, you said, "You win, Socrates," and I said, "Thank you, Meno," and then you said, "You're welcome," and then I said, "Do you admit there is no learning, only remembering?" and you answered, "I admit it." Would I have passed your challenge?
Meno: Certainly.
Socrates: Even though what you said began with "You win, Socrates," and not with "print"?
Meno: Yes, in this example the transformation has not directly produced my future conversation lines, but rather, a program which produces those lines. I would grant you victory for that. But again, that can never happen, for the same reason as before, Socrates.
Socrates: What about the following scenario? Suppose we describe a transformation, and you execute that transformation on this conversation thus far, from the beginning until the moment I said, "What about the following scenario?" And suppose the output to be:

```
socrates_line = input("Enter Socrates' next line:  ")
if socrates_line == "What do you think?":
print("I think you win, Socrates.")
else:
print("I think you lose, Socrates.")
```

And suppose that as soon as you saw that, I said, "What do you think?", and you answered, "I think you win, Socrates." Would I then have won your challenge?

Meno: Sure, Socrates. But once again, that would never happen, for the same reason as above.

Socrates: But what if, instead of my saying, "What do you think?", I instead said, "How now, Meno?", and then you said, "I think you lose, Socrates." Would that qualify for victory?

Meno: Yes.

Socrates: Why?

Meno: Because the transformation produced a program saying how I will respond to you. Namely, it said that if you next say "What do you think?", I will answer "I think you win, Socrates." And that otherwise, I'll say, "I think you lose, Socrates." And since you next say, "How now, Meno?", rather than "What do you think?", and since I then say "I think you lose, Socrates," the prediction was correct.

Socrates: But that would never happen, though, would it?

Meno: No, for I would intentionally defy the prediction. If you said, "What do you think?", I would calculate that the prediction is that I will say "I think you win, Socrates," and I would defy that prediction, and say "I think you lose, Socrates." But if you said, "How now, Meno?", then I would already know that the prediction is that I will answer "I think you lose, Socrates," and I would defy that prediction by saying "I think you win, Socrates." And we would both know I was lying, and that you really lost.

Socrates: You have really put me in a bind here, Meno. I fail to see any way for me to win, unless maybe some god appears from the air, some Deus ex Machina, as in the plays performed at the festival.

Meno: Do you give up then, Socrates?

Socrates: It feels almost hopeless, but I do have one ray of hope. It is not only me speaking in this conversation, but you too, Meno. So I will guide you to describe some transformation like we have talked about, and I pray that by some miracle our conversation just so happens to be exactly the right conversation that your transformation transforms into a program that predicts Meno, not through any wisdom of my own, for I believe I know nothing myself, but through your own mercy and your own wisdom, dear Meno, both of which so far exceed mine. Tell me, is it likely that there is some direct cipher which transforms our conversation into a good program? For instance, suppose the transformation says to take our conversation and change every "a" to "b", and change every "c" to "d", and so on, and finally change every "z" into "a". Is it at all likely this simple transformation could turn our conversation into a program that predicts you?

Meno: It would be an absolute miracle if that transformation turned our conversation into any valid program at all, Socrates. I'm nearly certain it would produce complete nonsense. Indeed, our conversation started with

"Meno:Goodmorning,Socrates."

That cipher you just proposed would yield an output starting with

"Nfop:Hppenpsojoh,Tpdsbufs."

I doubt any computer program starts like that. Much less one good enough to predict me!
Socrates: Perhaps the problem is that a haystack can't be made entirely out of needles, and if our conversation does secretly hide some program that predicts you, I suppose many characters from the conversation must work together to hide each character in the program. Perhaps every five characters from our conversation could be used to generate one character of our desired program. For example, perhaps the correct cipher, rather than changing "a" into "b", changes "Meno:" into "p", and changes "Goodm" into "r", and changes "ornin" into "i", and changes "g,Soc" into "n", and changes "rates" into "t", so that, altogether, the first 25 characters of the conversation become "print". Do you think that's at all likely?
Meno: It's more likely than it was before, but only marginally. It would still be quite a miracle if it worked, only slightly less of a miracle. I'm sure five to one is still way too many needles and too little hay, Socrates.
Socrates: Would ten characters of conversation per every one character of program be enough? That is, could our cipher transform, say, "Meno:Goodm" into "p", and "orning,Soc" into "r", and so on?
Meno: I'm fairly certain ten to one is still too little hay, Socrates.
Socrates: Well then, how many characters of conversation do you think would be enough, to encode one character of this elusive program we are looking for?
Meno: I don't really know. I guess it must depend on the size of the alphabet used in writing the program.
Socrates: You mean, if programs were written with a larger alphabet, then we would need more characters of conversation per character of the hidden program?
Meno: Yes.
Socrates: Then I guess since we haven't got the whole day to stand here talking, we should agree to write this desired program with the smallest alphabet we can. We should ask ourselves: which characters are absolutely essential for writing computer programs? Within reason, that is. After all, we're not trying to compete in some obfuscated code contest.
Meno: I think a reasonable alphabet for writing programs, without too many characters, would be:

```
alphabet = "abcdefghijklmnopqrstuvwxyz"
alphabet += " ()[]:'=,+.#\n"
alphabet += '"'
```

Socrates: That's forty characters in all, right? Since I assume by \n you mean a linebreak, which is one single character. And no uppercase letters, like "A", "B", "C"?

Meno: For your sake, Socrates, I shall consider it a victory for you if our program predicts I'll say "hey", all lowercase, even if my servant here actually writes "Hey" with the H uppercase.
Socrates: Most generous of you, Meno. Well then, if we assume this rather barebones alphabet of yours is rich enough to write computer programs, how many letters of our conversation should we use to encode each character of our program? We already saw that "one", "five", and "ten" are too small. Perhaps twenty?
Meno: Socrates, I know you haven't studied computer science, but really. Twenty is still way too small. You're probably still off by an order of magnitude, so let's make it two hundred.
Socrates: Alright, 200 it shall be. So every 200 characters of our conversation shall hide within them a single character of a computer program. Alright then, how would you define the cipher? It must take a string of 200 characters from our conversation and convert that string into a single character from your 40 letter alphabet.
Meno: Let me think, Socrates. Hmmm... Well, there's a convenient function named "ord" for turning characters into numbers. For example, ord("a") is 97, ord("b") is 98, ord("+") is 43, and so on. Don't fret about those specific numbers, this "ord" function is built into my programming language (Python). As our tablet has quotation marks and my servant sometimes writes those with fancy unicode quotes, I guess to be safe we should consider all the quotes in our conversation to be plain ASCII quotes, just like we delete all the spaces and linebreaks and tabs in the conversation. And plain ASCII quotes have ords as follows: for the single quote, ord("'")=39; for the double quote, ord('"')=34. With this ord function, we can systematically turn characters into numbers, add up those numbers, and then take the remainder after dividing the sum by the length of my program alphabet. In other words, if the sum is 40 or more, smash it down into the range from 0 to 39, so it can be used as the index of a character in our alphabet. If we write it all as a function:

```
def cipher(X):
total = sum(ord(character) for character in X)
remainder = total % len(alphabet)
return alphabet[remainder]
```

This function works for inputs X of any length. Of course it is intended to be used on inputs X of length 200.
Socrates: Just to make sure I understand right, what would be the value of, say, cipher("Meno:")?
Meno: Well, ord("M") is 77, ord("e") is 101, ord("n") is 110, ord("o") is 111, and ord(":") is 58. Add those up and you get 457. Now, the length of my alphabet is 40. If we divide 457 by 40, the result is 11 plus a remainder of 17. So the index we get is 457 % 40 = 17. So cipher("Meno:") is alphabet[17], character number 17 in our alphabet, remembering to count from zero. Which is "r". Altogether, cipher("Meno:")="r".

Socrates: Amazing, Meno, remind me someday to introduce you to my friend Timaeus. And what do we get when we take the first 200 characters from this conversation and plug them into this cipher?
Meno: If we take the first 200 characters of our conversation, they make the following string, though I fear my servant will insert linebreaks because otherwise this long string won't fit in one single row on his tablet, so please ignore those linebreaks; all that said, those 200 characters are as follows:

```
Meno:Goodmorning,Socrates.Socrates:Goodmorningtoyo
u,Meno,myfriend.Meno:I'vebeenwantingtotalktoyou.So
crates:Andjustwhatwouldyouliketotalkabout,mydearfr
iend?Meno:It'saboutwhatyousaidthelasttimeyouandIsp
```

If we plug this length 200 string into the cipher function, remembering not to include the linebreaks, we get the output "#".
Socrates: Great! And what about the next 200 characters?
Meno: The next 200 characters, in other words, characters number 200 through 399 (for computers start counting from zero), are as follows:

```
oketogether.Socrates:IsupposeyoumeanwhenItoldyouth
atlearningisreallyjustremembering?Meno:Yes,andthat
weactuallyknoweverythingalready,andwemerelyremembe
rit.Socrates:AndIsupposeyou'vecontrivedsomesortofp
```

And we can be confident we have not made a mistake, for the conversation flows seemlessly from the first 200 characters into the second 200 characters, you see? The first 200 characters ended with "sp", and the next 200 characters begin with "oke", and together that makes "spoke". If we plug these 200 characters into our cipher function, we get "t".
Socrates: So the computer program we are looking for must begin with "#t", is that right?
Meno: That's right, Socrates.
Socrates: Is it possible "#t" could be the beginning of a valid program?
Meno: Well, any line starting with "#" is a comment, which is meant for the human reader to read, not for the machine itself to run. The program could start with a comment, like "#this predicts meno" or "#this generates meno's next lines," or really any comment that begins with a "#t"; it's just a comment, it has no effect on what the program ultimately produces. So it's possible "#t" could be the start of a program.
Socrates: By Zeus! Alright, Meno, keep going. Plug in all the length 200 blocks from our conversation, stopping with the first block in which I say "Zeus". Do we get a valid program?
Meno: One minute... No, but I am shocked, for the output does resemble a computer program, but it has all kinds of errors. Here it is, note the double space between the "n" and the "r" on line 3:

```
#trojn hrse
seed(p=askpword())
z,x=", yo ooned","""n r''s'
iati.idnet'pas'r.p(se)d nt yap'o
+twspcw'tr"""
c=jn(smpl(p,z.len
shfl(d=rng(x.len
if c==z:exec(jn(x.i:i in d
```

Socrates: You say it has errors? Can we repair it? Tell me, what's the first error you notice?
Meno: The first line is spelled wrong. The comment should be "#Trojan Horse".
Socrates: I assume that's not the only error. Repair that. What's the next error?
Meno: The last three lines are missing two closing right parentheses each.
Socrates: If you repair that, what's the next error?
Meno: This code defines variables inside parentheses, as if the author were writing C instead of Python. For example, the line "seed(p=askpword())" should be two lines: "p=askpword()", then "seed(p)".
Socrates: Go ahead and repair that line. I see the same error in the definition of "d" in the penultimate line. Repair that, too. Are there any other errors?
Meno: The functions "seed", "smpl", and "shfl" are undefined.
Socrates: Does any library have those? Maybe we can import them.
Meno: Not by those exact names. Wait! The "random" library has "seed", "sample", and "shuffle", that must be what's meant. I'll insert a line "import random" right below the Trojan horse comment. And I'll change "seed" to "random.seed", "smpl" to "random.sample", and "shfl" to "random.shuffle".
Socrates: Any other errors?
Meno: "z.len" should be "len(z)" and "x.len" should be "len(x)".
Socrates: Anything else?
Meno: The expression "x.i:i in d" should clearly be "x[i] for i in d". It's like someone confused Python comprehensions with mathematical set notation.
Socrates: Anything else?
Meno: It looks like "rng" should be the standard function "range". And I think "jn" should be '"".join'.
Socrates: Anything else?
Meno: In modern Python, "range(len(x))" should be "list(range(len(x)))".
Socrates: Once you fix all that, are there any more errors?
Meno: Just one. This function "askpword" is undefined.
Socrates: Could the name be short for something?
Meno: Maybe "ask_password". I'll replace that line with "p=input('Enter the password: ')".
Socrates: What's the program now, with all the errors fixed?
Meno: Here you go:

```
#Trojan Horse
import random
p=input('Enter the password:  ')
random.seed(p)
z,x=", yo ooned","""n r''s'
iati.idnet'pas'r.p(se)d nt yap'o
+twspcw'tr"""
c="".join(random.sample(p,len(z)))
d=list(range(len(x)))
random.shuffle(d)
if c==z:exec("".join(x[i] for i in d))
```

Socrates: Run it! What does it predict? I understand you plan to say the opposite. What's wrong? You are quiet. Is there some problem with your program?
Meno: It wants a password.
Socrates: You're in a bind, for until you enter the password, you won't know what it will say once you enter it. Maybe when you enter the password, it will say, "It wants a password." It's too late to say the opposite of that. Maybe studying the code, you can figure out the next prediction, and defy the oracle? What's wrong? Can't you analyze the code to determine the next prediction?
Meno: It's encrypted.
Socrates: Then you must enter the password, and we shall see what your program predicted. Meno! Tell me! What is the password?
Meno: I don't know any password, Socrates.
Socrates: The password, in your own words, is: "I don't know any password, Socrates." Enter that, and if your wisdom is as great as I believe, I think your program will print:

```
it wants a password.
it's encrypted.
I don't know any password, Socrates.
```

Now I must leave, Meno, I have an appointment at the courthouse.

3 Conclusion

One moral we might draw from this story is that the statement "philosophy will help me remember future contingent facts even if I try to contradict them" could itself be contingent. Meno's paradoxical challenge initially seems to disprove this statement by pure reasoning (like the geometry example in Plato's *Meno*), which would make the statement universally false. Socrates' victory disproves that disproof.

We wrote the dialogue using annotations. For example, "Meno: [Yes|Correct]" means that Meno could say "Yes" or "Correct". A computer program flattened these annotations to encode Meno's program. The ways a passage can be so obtained grow exponentially with the number of annotations. "Meno: [Yes|Indeed], [Socrates|friend]" can be flattened in 4 ways; "Socrates: [Isn't that|Is that not] [right|correct], [Meno|friend]?" can be flattened in 8 ways. Thus the problem is easier than it initially appears. The annotated dialogue, and associated computer programs, are available at `https://github.com/semitrivial/Meno`.

As an application, a variation of this technique could theoretically be used to assist in the problem of preserving rhyme and metre when translating poetry. We speculate perhaps similar methods were used (without computer assistance) for example by John Dryden, whose English translation of Virgil's Aeneid rhymes almost every line.