AI Turing Test

Loops, Decisions, Simple String Parsing, Intro to Turing's AI Theory, Abstraction

Due: ______________________________ due date

Part I: Building Q&A

Ask Questions to the Given Teachers

This homework will introduce you to Turing's theory of artificial intelligence through a simple version of his most well-known work: the Turing Test. This test is designed as a Q&A game where one participant tries to determine if the other one is a human or a robot based on their responses. For this homework, we'll be doing a very simple version of this game.

You're job in this part is to build the Q&A platform for you Turing Test. We have provided you with three classes: "Human" which will represent the human teachers; "Robot" which will represent the robot teacher; and "District" which will represent the whole district. "Human" and "Robot" objects have the same interface, so you won't be able to know if you're interacting with a Human or a Robot based on the code alone. We'll talk about Districts more in the next part.

You are provided with a list of Districts and a list of questions to ask the Teachers. For this part, you are to loop through each District. In each District, you are to loop through each Teacher. For each Teacher, you are to display their name and ask them each question and display their response. These are the methods at your exposure for Teachers:

- getFirstName(): Returns a String of the Teacher's first name
- getLastName(): Returns a String of the Teacher's last name
- ask(question): Takes in a String and returns the Teacher's answer to the question. If the Teacher doesn't have an answer, they'll respond that they don't know
- __str__(): This function gets called with str(obj) or print(obj). It returns the Teacher's name and what they are (Human or Robot). ONLY USE THIS AT THE END, NOT FOR THE Q&A SECTION

To use any of these functions, take a Teacher object (let's say t) and call $t.method(argument)$. To access the teachers for District d, loop through $d.teachers$.

Part II: Turing Test

Build User Interface and Parse Results

Now to build the test itself! Now that you have the questions and responses, you want your user to guess whether or not the Teacher is a Human or a Robot. How you choose to do this is up to you, but you need to track their responses in a list so we can check them later. You'll notice that there are a LOT of Teachers to go through. Once you feel like your input and output are working, you can automate this process using random.choice if you'd like, but don't delete your input code (simply comment it out).

Once you've collected your responses, you'll need to calculate the budget for the District you're investigating. In this case, you can simply use a list and assume that the index in the *district* list will match up with the index of the *budget* list (e.g. if ”Seattle” is index 2, ”Seattle”s budget will also be index 2). Here are the rules for budgeting:

- A Human costs $25,000 per year
- A Robot costs $15,000 per year

You will be building this budget based on what the user guessed from the Turing Test, so your final budgets might not be accurate. We'll check which ones were right and which ones were wrong.

Each District comes pre-loaded with a known budget. For each District, you will print one of the following statements:

- If the calculated budget matches the known budget: Print ”We have enough money to pay everyone a fair wage in [District.name]”
- If the calculated budget is less than the known budget: Print ”The [District.name] district will be underfunded by $[difference//1000],000 next year”
- If the calculated budget is more than the known budget: Print ”We do not have the funds for the [District.name] district budget. We will have a debt of $[difference//1000],000”

At the end, if you're curious what the actual budget was or what the Teachers were, you can call *print*(d) for some District d to see all information about the District.

Part III: README

Final Thoughts

Please respond to the questions in the provided README.txt file and submit it along with your code.