Name: **Trisha Ganesh**

Date: **6/27/2021**

Magpie Chatbot Lab

Directions: Make note of your responses to the following questions as you work through activities 4 and 5 of the AP Computer Science Lab Student Guide: Magpie.

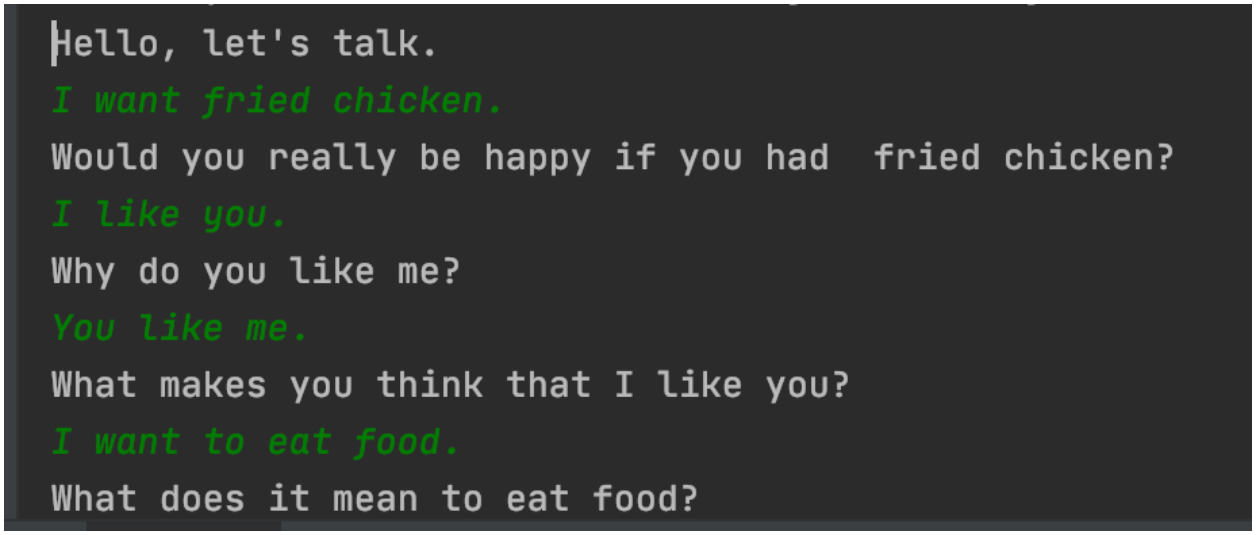
Activity 4

1. In the Exploration section, how does the Magpie chatbot respond to:

I want to build a robot.	**Response:** **"*What would it mean to build a robot?"***
I want to understand French.	**Response:** **"*What would it mean to understand French?"***
Do you like me?	**Response:** **"*What makes you think that I like you?"***
You confuse me.	**Response:** **"*What makes you think that I confuse you?"***

1. After altering the code, test the code by chatting with Magpie. Test statements in the form of "I want to *something*", "I want *something*", "you *something* me", and "I *something* you". Paste your conversation below.

Here is my conversion with Magpie:



1. **When altering the code, why did you need to be careful about where you place the check for "I want *something*" statements? Use the samples to determine your answer.**

When altering the code, I needed to be vary of how my edits would have an effect on the Magpie's responses. For instance, my goal was to find the keyword(s) *"I want "* in the user's statement. Therefore, I needed to ensure that my goal was set to "I want", in my program and the starting position was set to zero to ensure the user's statement starts with those specific keywords. For the user's statement *"I want fried chicken",* I needed to ensure that the return statement (Magpie's response was accurate). "Would you really be happy if you had" + restOfStatement + "?". [restOfStatement = fried chicken]. The program also takes into account the important keyword(s): "I want" (in my case), in order to successfully print the message: "Would you really be happy if you had fried chicken?"

Activity 5

1. **Imagine you were to interview an end user and ask them to compare the version of the program before the array was added to `getRandomResponse` and the version after. What might they say?**

The end user may agree that the addition of the array might've been a more efficient way to write the program. Before, with the use of if, else-if statements, our program was long and might've been difficult to comprehend. However, with the use of the String array, we were able to add all the random responses into a single array, making it easier for anyone to understand. The second approach is more straightforward. Before, the use of "whichResponse == 0, whichResponse == 1, whichResponse == 2, whichResponse == 3, (etc.) inside our if statements, may have been confusing to understand, especially for a user with little programming experience. Therefore, the use of the string array, clearly displays all the random responses thrown back to a user if a certain type of statement is entered.

1. **Now interview the programmer for the code. How might he/she compare and contrast the two programs? Are the end user's and programmer's responses similar? Why?**

The programmer may also have the same belief as the end user. Programmers want to write clean, concise code, making it easy for humans to read. With the use of a String Array, the programmer may agree that is an efficient way to return random responses to the user, while still retrieving a similar output of random responses to certain statements. In addition, the programmer will agree that using a String Array is a good way to utilize arrays, especially for a program like this. Programmers want to explore new concepts and understand how they can be utilized in a variety of programs. Therefore, the end user and the programmer will have similar opinions for the two programs.