

I. My information:

// Course: 3642/01

// Student name: Andujar Brutus

// Student ID: 001002015

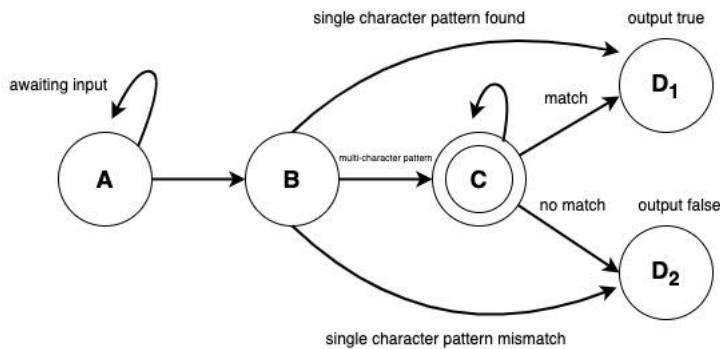
// Assignment #: 1

// Due Date: September 16th 2023

// Signature: Andujar Brutus (Your signature assures that everything is your own work. Required)

// Score: _____ (Note: Score will be posted on D2L)

II. Design of my Agent:



I used Spring and React to design an agent that can be integrated into a gui and axios.js to connect the backend to the frontend.

CounterController.java – The primary backend class

```
package com.example.aiproject1;

import org.springframework.web.bind.annotation.*;

@CrossOrigin(origins = "http://localhost:3000/")
@RestController
@RequestMapping(value = "/counter")
public class CounterController {
    //instantiate the inputs as static
    static String sentenceInput;
    static String patternInput;

    @CrossOrigin
    @PostMapping("/pattern/{pattern}")
    public String enterPattern(@PathVariable String pattern) {
        //prompt user for input
        patternInput = pattern;
        return "Pattern set to " + patternInput;
    }

    //primarily for testing
    @PostMapping("/pattern")
    public String showPattern() {
        //return the pattern
        return "Current pattern is " + patternInput;
    }

    @CrossOrigin
```

```

@PostMapping("/sentence/{sentence}")
public String enterSentence(@PathVariable String sentence) {
    //enter a string to be searched
    sentenceInput = sentence;
    return "Sentence set to " + sentenceInput;
}

//mainly used for testing
@PostMapping("/sentence")
public String showSentence() {
    //return the current sentence
    return "Current sentence is " + sentenceInput;
}

@CrossOrigin
@GetMapping("/searchsentence")
public String searchSentence() {
    //search for the pattern in the string
    if (sentenceInput.contains(patternInput)) {
        return ("I found the pattern '" + patternInput + "' in the sentence '" +
sentenceInput + "'");
    } else {
        return ("The pattern '" + patternInput + "' doesn't exist in the sentence '" +
sentenceInput + "'");
    }
}
}

```

PatternAgent.jsx – the primary frontend class

```

import api from "../api/axiosConfig";

const PatternAgent = () => {
    //variables to store input
    let patternIn = "";
    let sentenceIn = "";

    //functions to be used in GUI

    //store the pattern from gui form and send to api
    async function savePattern(event) {
        patternIn = event;
        if (patternIn == null || patternIn == " " || patternIn == ""){
            patternIn = "none"
        }
        await api.post("/pattern/" + patternIn)
        console.log(patternIn);
    }

    //store the sentence from GUI form and send to api
    async function saveSentence(event) {
        sentenceIn = event;
        if (sentenceIn == null || sentenceIn == " " || sentenceIn == ""){
            sentenceIn = "none"
        }
        await api.post("/sentence/" + sentenceIn)
        console.log(sentenceIn);
    }
}

```

```

//execute the search
async function searchSentence(event) {
  await api.get("/searchsentence").then(function (response) {
    alert(response.data);
  });
}

// forms to accept GUI input and search button
return (
  <div className="container mt-4">
    <form>
      <div className="form-group my-2">
        <label>Pattern</label>
        <input
          type="text"
          className="form-control"
          placeholder="Enter a pattern for the AI agent to find..."
          onChange={e => savePattern(e.target.value)}
        />
      </div>

      <div className="form-group mb-2">
        <label>Sentence</label>
        <input
          type="text"
          className="form-control"
          placeholder="Enter a sentence in which to find the pattern..."
          onChange={e => saveSentence(e.target.value)}
        />
      </div>

      <div>
        <button className="btn btn-primary m-4" onClick={searchSentence}>
          Search
        </button>
      </div>

    </form>
  </div>

);
};

export default PatternAgent;

```

III. Tasks that my agent will solve:

My agent can match patterns with strings that are part of a sentence. It can match strings of any size as long as the pattern exists to be matched.

Pattern Searching AI Agent

By Andujar Brutus

Pattern

1234

Sentence

1234

Search

I found the pattern '1234' in the sentence '1234'

Close

Pattern Searching AI Agent

By Andujar Brutus

Pattern

KSU2023

Sentence

This sentence does not contain the pattern

Search

The pattern 'KSU2023' doesn't exist in the sentence 'This sentence does not contain the pattern'

Close

Pattern Searching AI Agent

By Andujar Brutus

Pattern

Sentence

Search

I found the pattern 'none' in the sentence 'none'

[Close](#)

Pattern Searching AI Agent

By Andujar Brutus

Pattern

Sentence

Search

The pattern 'some' doesn't exist in the sentence 'none'

[Close](#)