

CSC341 HW1

Havin Lim

February 6th 2024

1 Academic Honesty

Written Sources Used: Michael Sipser - Introduction to the Theory of Computation
C# - if, else if, else Statements - <https://www.tutorialsteacher.com/csharp/csharp-if-else>

Help Obtained: None

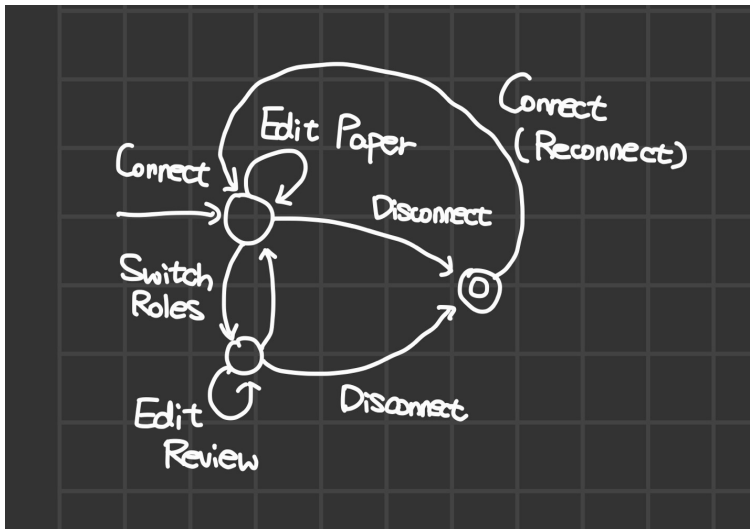
2 Exercises

2.1 Problem 1

$\Sigma = \{\text{Connect, Switch Roles, Edit paper, Edit review, Disconnect}\}$

$L = \{\mathbf{w} \in \Sigma^* \mid \text{String } \mathbf{w} \text{ follows the rules for sessions between the client and the server}\}$

2.2 Problem 2



2.3 Problem 3

Author State: Initial connection state, connects as the author role.

Property: The client was not yet connected or has performed a "Switch Roles" command.

Case Analysis:

If the command is "Edit Paper," it allows the client to edit the paper and remain in the same state. This transition is valid and preserves the property since the client is allowed to edit papers in this state.

If the command is "Switch Roles," transition to the "Reviewer State". This transition is valid and preserves the property since it shows that the client has changed roles from author state to reviewer state.

If the command is "Disconnect," transition to the "End Session State". This transition is valid and preserves the property as the session ends.

Any other command leads to a dead state. No other command is valid.

Reviewer State: Connects as the reviewer role.

Property: The client has performed a "Switch Roles" command from the author role.

Case Analysis:

If the command is "Edit Review," it allows the client to edit the paper and remain in the same state. This transition is valid and preserves the property since the client is allowed to review papers in this state.

If the command is "Switch Roles," transition to the "Author State". This transition is valid and preserves the property since it shows that the client has changed roles from reviewer state to author state.

If the command is "Disconnect," transition to the "End Session State". This transition is valid and preserves the property as the session ends.

Any other command leads to a dead state. No other command is valid.

End Session State: Disconnects and ends the current session.

Property: The client has performed a "Disconnect" command while connected.

Case Analysis:

If the command is "Connect(Reconnect)" it allows the client to connect with a new session with the author role. This transition is valid and preserves the property since the client is allowed to connect again with a new session.

2.4 Problem 4

```
public class CMS
{
    private bool connection;
    private string role;
```

```

public CMS()
{
    connection = false;
    role = "author";
}

public void Connect()
{
    if (!connection)
    {
        Console.WriteLine("Connected to the conference management server.");
        connection = true;
    }
    else
    {
        Console.WriteLine("Error: Session is currently connected.");
    }
}

public void SwitchRoles()
{
    if (connection && role == "author")
    {
        role = "reviewer";
        Console.WriteLine("Role changed to reviewer.");
    }
    else if (connection && role == "reviewer")
    {
        role = "author";
        Console.WriteLine("Role changed to author.");
    }
}
else
Console.WriteLine("Error: Could not perform command.");
}

public void EditPaper()
{
    if (connection && role == "author")
    {
        Console.WriteLine("Start editing paper.");
    }
    else if (role == "reviewer")
    {
        Console.WriteLine("You are not authorized to edit papers as a reviewer.");
    }
    else

```

```

        {
            Console.WriteLine("Error: Could not perform command.");
        }
    }

    public void EditReview()
    {
        if (connection && role == "reviewer")
        {
            Console.WriteLine("Start editing review.");
        }
        else if (role == "author")
        {
            Console.WriteLine("You are not authorized to edit papers as an author.");
        }
        else
        {
            Console.WriteLine("Error: Could not perform command.");
        }
    }

    public void Disconnect()
    {
        if (connection)
        {
            Console.WriteLine("Disconnected from the conference management server.");
            connection = false;
            role = "author";
        }
        else
        {
            Console.WriteLine("Error: Could not perform command.");
        }
    }
}

```

2.5 Problem 5

It is profitable to construct finite automata before implementation because it is easier for the programmer to see how the user can interact with the program, in this case, the CMS. On the other hand, if the program, or the number of commands and actions that could be performed by the user, gets complicated or grows in number, it would be complex and difficult to create a finite automata. It would be time-consuming and difficult to understand while one is creating it.