**­**

|  |  |
| --- | --- |
| **Total Marks:** | **04** |
| **Obtained Marks:** |  |

**Finite Automata Theory and Formal Languages**

**Assignment # 03**

**Last date of Submission: 5 Dec 2024**

**Submitted To: Muhammad Nadeem Khokhar**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Student Name: UBAID-BIN-WARIS**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Reg Number: 2212416**

***Instructions****: Copied or shown assignments will be marked zero. Late submissions are not entertained in any case.*

**Question**

A Moore machine is provided with four states: q0, q1, q2, and q3. The machine begins in the initial state, q0, and generates outputs based on its current state. The output is 1 when the machine enters either q0 or q3, and 0 when it enters q1 or q2. State transitions depend on the input symbol read, where input 'a' leads to a different state than input 'b'. The specific state transitions are as follows.

* Reading 'a' at q0 takes you to q1
* Reading 'b' at q0 takes you to q3
* Reading 'a' at q1 takes you to q3
* Reading 'b' at q1 takes you to q1
* Reading 'a' at q2 takes you to q0
* Reading 'b' at q2 takes you to q3
* Reading 'a' at q3 takes you to q3
* Reading 'b' at q3 takes you to q2

Your task is to:

1. Create a transition table using MS Word.
2. Design the transition diagram using JFlap.
3. Implement the Moore machine in C/C++. The implementation should ensure that the machine starts from the initial state only on its first run and resumes from its last state on subsequent executions.
4. Test the implemented machine by running the input strings 'abba,' 'abb,' and 'ba.' Provide the complete runtime output screen for these test cases.

**Note:**

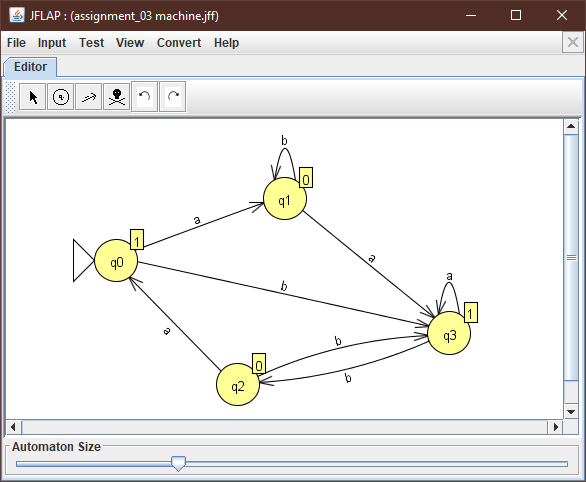
1. Change the filename to your ID, e.g. 2073105.doc
2. Upload the .doc on Google Classroom.
3. Do not use system calls.
4. Make sure that the output screen is complete and does not have colored or black background.
5. Poor indentation and wrong format will result in deduction of marks.

**Solution**

**Transition table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Old State** | **New State** | | | |
| **a** | | **b** | |
| **State** | **Output** | **State** | **Output** |
|  |  | 0 |  | 1 |
|  |  | 1 |  | 0 |
|  |  | 1 |  | 1 |
|  |  | 1 |  | 0 |

**Transition diagram using JFlap**



**Code**

#include <iostream>

#include<fstream>

using namespace std;

string input;

int ctr = 0;

char value;

int current\_state;

void state\_q0();

void state\_q1();

void state\_q2();

void state\_q3();

void load\_state();

void select\_state();

void update\_state();

int main()

{

while (true) {

cout << "Enter 'exit' to terminate the program! \nEnter a String: ";

cin >> input;

if (input == "exit") {

cout << "\n\n\t\t\t > > > Exiting The Program < < < " << endl;

return 0;

}

ctr = 0;

load\_state();

select\_state();

update\_state();

}

}

void select\_state()

{

switch (current\_state)

{

case 0:

state\_q0();

break;

case 1:

state\_q1();

break;

case 2:

state\_q2();

break;

case 3:

state\_q3();

break;

default:

cout << "Invalid State" << endl;

break;

}

}

void state\_q0()

{

cout << "1" << endl;

current\_state = 0;

if (ctr >= input.length())

{

cout << "\tString Accepted > > >" << endl;

return;

}

value = input[ctr++];

if (value == 'a')

state\_q1();

else if (value == 'b')

state\_q3();

else

cout << "Invalid String Returning ! ! !" << endl;

}

void state\_q1()

{

cout << "0" << endl;

current\_state = 1;

if (ctr >= input.length())

{

cout << "\tString Accepted > > >" << endl;

return;

}

value = input[ctr++];

if (value == 'a')

state\_q3();

else if (value == 'b')

state\_q1();

else

cout << "Invalid String Returning ! ! !" << endl;

}

void state\_q2()

{

cout << "0" << endl;

current\_state = 2;

if (ctr >= input.length())

{

cout << "\tString Accepted > > >" << endl;

return;

}

value = input[ctr++];

if (value == 'a')

state\_q0();

else if (value == 'b')

state\_q3();

else

cout << "Invalid String Returning ! ! !" << endl;

}

void state\_q3()

{

cout << "1" << endl;

current\_state = 3;

if (ctr >= input.length())

{

cout << "\tString Accepted > > > " << endl;

return;

}

value = input[ctr++];

if (value == 'a')

state\_q3();

else if (value == 'b')

state\_q2();

else

cout << "Invalid String Returning ! ! !" << endl;

}

void load\_state()

{

ifstream file("states.txt");

if (!file.good())

{

ofstream outfile("states.txt");

outfile << "0" << endl;

outfile.close();

current\_state = 0;

cout << "\t\t File created successfully with initial state 0" << endl << endl;

}

else

file >> current\_state;

file.close();

}

void update\_state()

{

ofstream file("states.txt");

file << current\_state << endl;

file.close();

}

**Output**

