Compiler Design Lab (CS 306L)

Katakam Akshay Reddy

AP19110010425

CSE-C

Week 1: Implementation of Language recognizer

Week 1 Programs

1. Implementation of Language recognizer for set of all strings over input alphabet ∑={a,b} containing even number of a's and even number of b's.
2. Implementation of Language recognizer for set of all strings ending with two symbols of same type.

Program 1:

Implement a language recogniser which accepts set of all strings over the alphabet ∑={a,b} containing an even number of a's and an even number of b's.

Description:

The acceptable strings of the language are ε(Null string), aa, bb, abba, babbab etc.

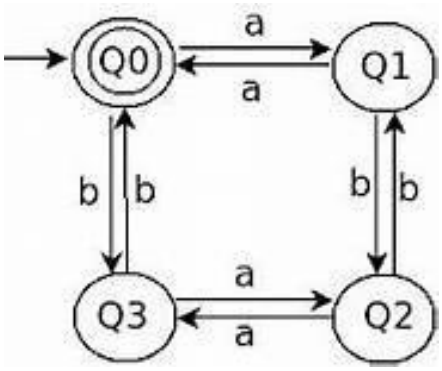
Deterministic Finite Automata for the given language is given below:

DFA M=(Q,∑,δ,Q_0,F) Where

Q=Set of all states ={Q_0,Q_1,Q_2,Q_3}

∑=Input Alphabet={a,b},

Start state is Q_0
F=Set of all final States={ Q_0}



And the transitions are defined in the transition diagram

Algorithm: Language recognizer

Input:

input //input string

Output:

Algorithm prints a message

"String accepted": If the input is acceptable by the language,

"String not accepted" otherwise,

"Invalid token": If the input string contains symbols other than input alphabet.

Method:

```
state=0   //initial state
i=0
while((current=input[i++])!='\0'){
    switch(state)          case 0:
if(current=='a')          state=1;
else if(current=='b')    state=2;
else
                          Print "Invalid token" ; exit;
            case 1: if(current=='a')
state=0;                  else if(current=='b')
state=3;                        else
                          Print "Invalid token" ; exit;
        case 2: if(current=='a')          state=3;
                else if(current=='b')     state=0;
          else
                          Print "Invalid token" ; exit;
            case 3: if(current=='a')
state=2;                   else if(current=='b')
state=1;                        else
                          Print "Invalid token" ;
exit;                                end switch     end
while //Print output if(state==0)
      Print ”String accepted” else
      Print ”String not accepted”
```

C Code:

```c
#include <stdio.h>
#include <stdlib.h>
void main()
{
    int state = 0, i = 0;
    char token, input[20];
    printf("Enter input string \t :");
    scanf("%s", input);
    //printf("Given string is : %s");

    while ((token = input[i++]) != '\0')
    {
        // printf("current token : %c \n",token);
        switch (state)
        {
        case 0:
            if (token == 'a')
                state = 1;
            else if (token == 'b')
                state = 2;
            else
            {
                printf("Invalid token");
                exit(0);
            }
            break;
        case 1:
            if (token == 'a')
                state = 0;
            else if (token == 'b')
                state = 3;
            else
            {
                printf("Invalid token");
                exit(0);
            }

            break;
        case 2:
            if (token == 'a')
                state = 3;
            else if (token == 'b')
                state = 0;
            else
            {
                printf("Invalid token");
                exit(0);
            }
            break;
        case 3:
            if (token == 'a')
                state = 2;
            else if (token == 'b')
                state = 1;
            else
            {
                printf("Invalid token");
                exit(0);
            }
            break;
        }
        // printf("state = %d  ",state);
    }
    if (state == 0)
        printf("\n\nString accepted\n\n");
    else
        printf("\n\nString not accepted\n\n");
}
```

Test cases:

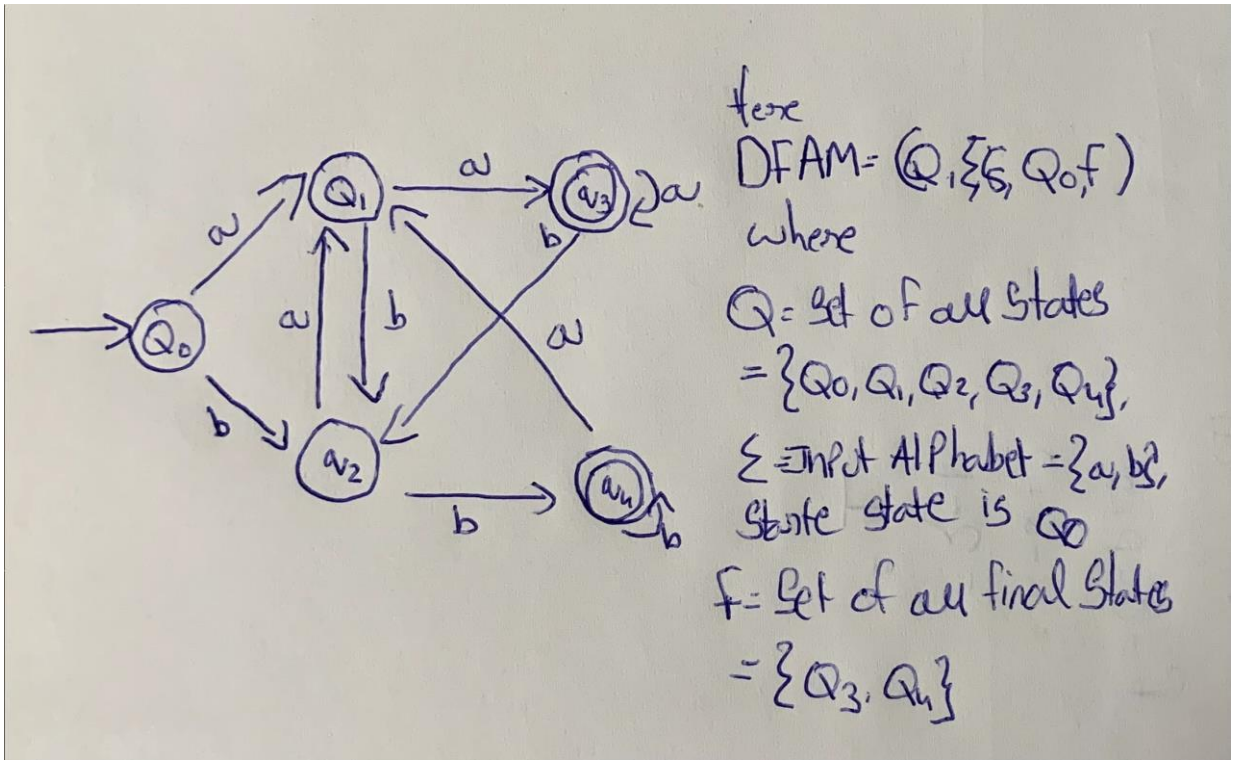
Input	Expected Output
aabb	String accepted
abab	String accepted
aaabb	String not accepted
aaa	String not accepted
abcd	Invalid token

Program 2 :

Implementation of Language recognizer for set of all strings ending with two symbols of same type.

Description :

Any string where the last two symbols were the same is acceptable. The strings are like aa, aaa, baa, bababb, etc. Deterministic Finite Automata for the given language is given below :



The transitions are defined in the transition diagram

Algorithm: Language recognizer

Input:

input //input string

Output:

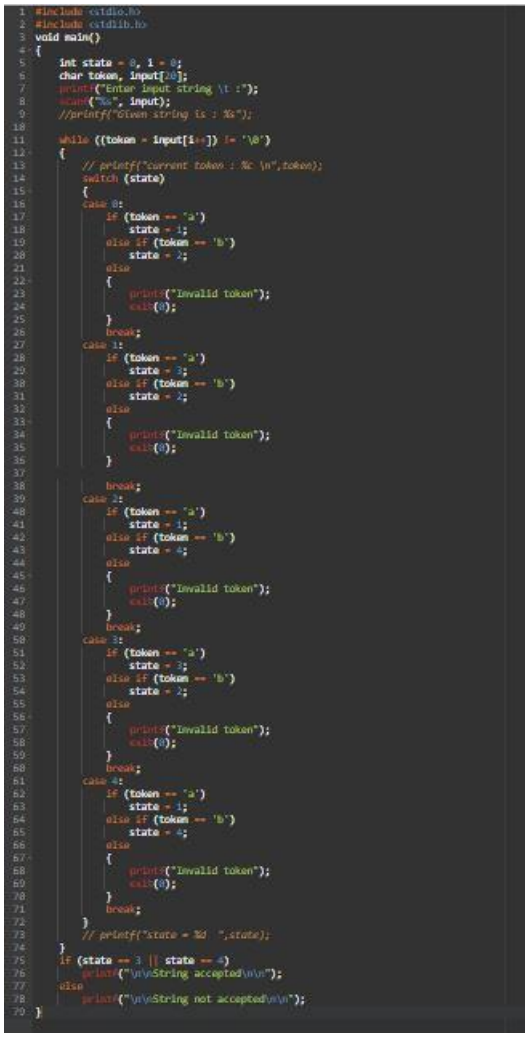
Algorithm prints a message

"String accepted": If the input is acceptable by the language,

"String not accepted" otherwise,

"Invalid token": If the input string contains symbols other than input alphabet.

C Code:



```c
#include <stdio.h>
#include <stdlib.h>
void main()
{
    int state = 0, i = 0;
    char token, input[20];
    printf("Enter input string \t :");
    scanf("%s", input);
    //printf("Given string is : %s");

    while ((token = input[i++]) != '\0')
    {
        // printf("current token : %c \n",token);
        switch (state)
        {
        case 0:
            if (token == 'a')
                state = 1;
            else if (token == 'b')
                state = 2;
            else
            {
                printf("Invalid token");
                exit(0);
            }
            break;
        case 1:
            if (token == 'a')
                state = 3;
            else if (token == 'b')
                state = 2;
            else
            {
                printf("Invalid token");
                exit(0);
            }

            break;
        case 2:
            if (token == 'a')
                state = 1;
            else if (token == 'b')
                state = 4;
            else
            {
                printf("Invalid token");
                exit(0);
            }
            break;
        case 3:
            if (token == 'a')
                state = 3;
            else if (token == 'b')
                state = 2;
            else
            {
                printf("Invalid token");
                exit(0);
            }
            break;
        case 4:
            if (token == 'a')
                state = 1;
            else if (token == 'b')
                state = 4;
            else
            {
                printf("Invalid token");
                exit(0);
            }
            break;
        }
        // printf("state = %d  ",state);
    }
    if (state == 3 || state == 4)
        printf("\n\nString accepted\n\n");
    else
        printf("\n\nString not accepted\n\n");
}
```

Test cases:

Input	Expected Output
aabb	String accepted
abab	String not accepted
aaabb	String accepted
aaa	String accepted
abcd	Invalid token