



Compiler Design Lab (CS 306L)

Week 1: Implementation of Language recognizer

Name: Kamisetty Sai Smaran

Roll No. AP19110010350

Section: CSE-C

Week 1 Programs:

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

Program 1:

Implement a language recogniser which accepts set of all strings over the alphabet $\Sigma = \{a, b\}$ 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 $M = (Q, \Sigma, \delta, Q_0, F)$ Where

$F = \text{Set of all final States} = \{ Q_0 \}$

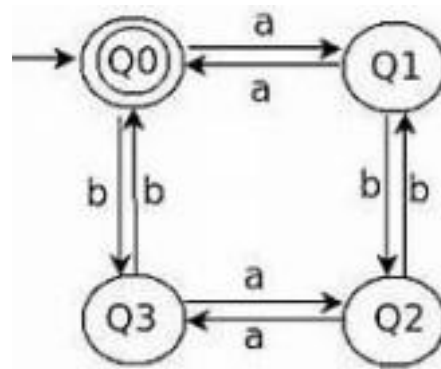
$Q = \text{Set of all states} = \{ Q_0, Q_1, Q_2, Q_3 \}$

And the transitions are defined in

$\Sigma = \text{Input Alphabet} = \{a, b\},$

the transition diagram

Start state is Q0



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:

```
#include<bits/stdc++.h>
using namespace std;

int main()
{
    int state=0;
    int i,j;
    char string[50]="aabb";

    for(i=0;i<4;i++){

        if (state==0){

            if (string[i]=='a'){
                state=1;

            }
            else if (string[i]=='b'){
                state=2;
            }
            else{

                cout<<"invalid entry";
                break;
            }

        }

        else if (state==1){
            if (string[i]=='a'){
                state=0;
            }
            else if (string[i]=='b'){
```

```

        state=3;
    }
    else{
        cout<<string[i];
        cout<<"invalid entry";
        break;
    }
}

else if (state==2){
    if (string[i]=='a'){
        state=3;
    }
    else if (string[i]=='b'){
        state=0;
    }
    else{
        cout<<"%c",string[i];
        cout<<"invalid entry";
        break;
    }
}

else if (state==3){
    if (string[i]=='a'){
        state=2;
    }
    else if (string[i]=='b'){
        state=1;
    }
    else{
        cout<<"invalid entry";
        break;
    }
}

}
if (state==0){
    cout<<"accepted";
}
else{
    cout<<"%d",state;
}

```

```

        cout<<"not accepted";
    }
}

```

Test cases:

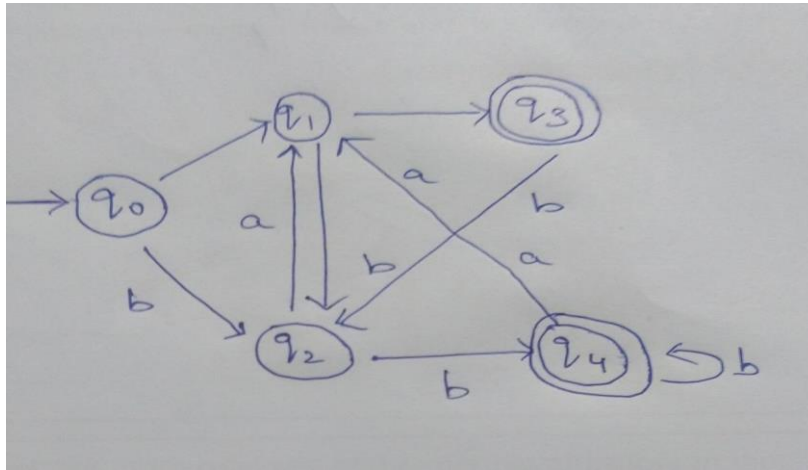
Input	Expected Output
aabb	String accepted
abab	String accepted
aaabb	String not accepted
aaa	String not accepted
abcd	Invalid token

Problem 2:

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

Description:

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



DFA $M=(Q,\Sigma,\delta,Q_0,F)$ Where

Q =Set of all states

$=\{Q_0,Q_1,Q_2,Q_3,Q_4\}$,

Σ =Input Alphabet= $\{a,b\}$,

Start state is Q_0

F =Set of all final States= $\{Q_3,Q_4\}$

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 the input alphabet.

Method:

state=0 //initial state i=0 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");
    }

```

C++ code:

```

#include<bits/stdc++.h>
using namespace std;

int main(){
    int i=0,initial = 0,n,state=0;
    char current,input[20];
    cout<<"Enter value of string(only a's and b's): ";
    cin>>input;
    while((current=input[i++])!='\0'){
        switch(state){
            case 0:
                if(current=='a'){
                    state=1;
                }
                else if(current=='b'){
                    state=3;
                }
                else{
                    cout<<"Invalid Input!";
                    exit(0);
                }
                break;
            case 1:
                if(current=='a'){
                    state=1;
                }
                else if(current=='b'){
                    state=2;
                }
                else{
                    cout<<"Invalid Input";
                    exit(0);
                }
                break;
            case 2:

```



```

        if(current=='a'){
            state=1;
        }
        else if(current=='b'){
            state=2;
        }
        else{
            cout<<"Invalid Input";
            exit(0);
        }
        break;
    case 3:
        if(current=='a'){
            state=4;
        }
        else if(current=='b'){
            state=3;
        }
        else{
            cout<<"Invalid Input";
            exit(0);
        }
        break;
    case 4:
        if(current=='a'){
            state=4;
        }
        else if(current=='b'){
            state=3;
        }
        else{
            cout<<"Invalid Input";
            exit(0);
        }
        break;
    }
}
if(state==1 || state==3){
    cout<<"String Accepted!!"<<endl;
}
else{
    cout<<"String Not Accepted!!"<<endl;
}
return 0;
}

```

Test cases:

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