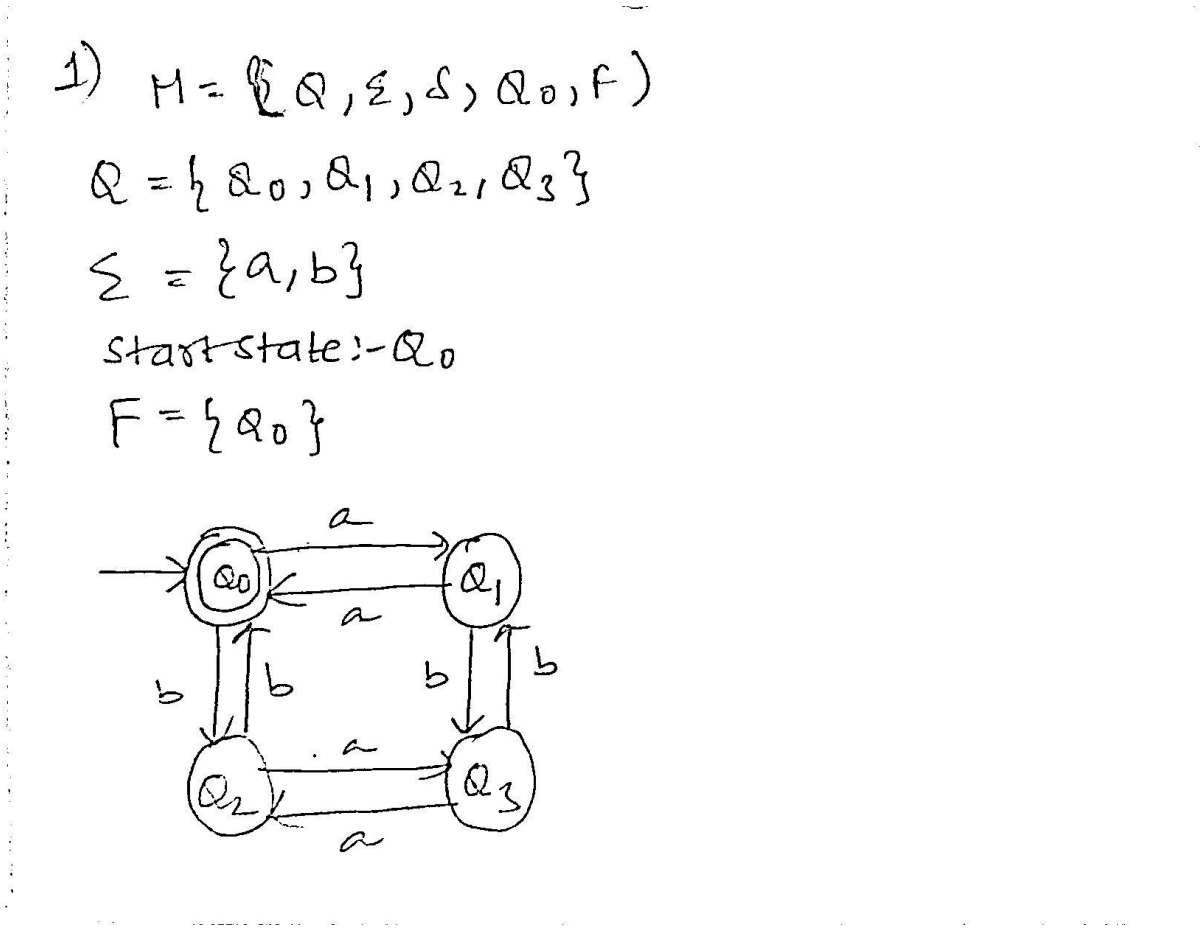
1)Implementation of Language recognizer for a set of all strings over input alphabet ∑={a,b} containing even number of a's and even number of b's.



Source code link:- **https://onlinegdb.com/kpHpETFVJ**

Program:-

```
#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");
}
```

Input	Output
aaabb	String not accepted
aabb	String accepted
abcd	Invalid token

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

2)

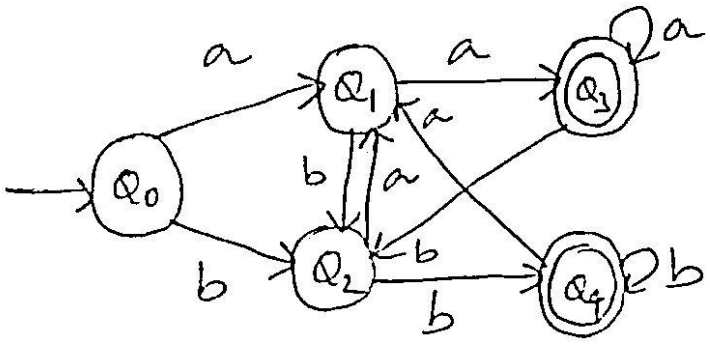
$M = (Q, \Sigma, \delta, Q_0, F)$

Q = set of all states $\{Q_0, Q_1, Q_2, Q_3, Q_4\}$

$\Sigma = \{a, b\}$

start state is Q_0

$F = \{Q_3, Q_4\}$



Source code link:- https://onlinegdb.com/jfUTaa5PB

Program:-

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
void main()
{
   int state=0,i=0;
   char token,input[20];
   printf("Enter input string \t :");
   scanf("%s",input);
   while((token=input[i++])!='\0')
   {
      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;
        }
    }
    if(state==3 || state==4)
        printf("\n\nString accepted\n\n");
    else
        printf("\n\nString not accepted\n\n");
}
```

Input	**Output**
abaaba	String not accepted
ababaaa	String accepted
abcaa	Invalid token

Output:-

```
Enter input string      :abaaba
String not accepted

Enter input string      :ababaaa
String accepted

Enter input string      :abcaa
```

Invalid token