

1. Problema

Codificar los siguientes autómatas por Diagrama de Transiciones y por Tabla de Transiciones:

- Autómata que verifique la correcta sintaxis de un identificador.
- Autómata que verifique la correcta sintaxis de un número real.

2. Código

2.1. Funciones Adicionales

Las funciones que se muestran a continuación son utilizadas para saber si un char constante o el contenido de un iterador de string, es un número o una letra.

```
1
2 bool esNumero(const char &caracter){
3     if(caracter >= 48 and caracter <= 57) return true;
4     return false;
5 }
6
7 bool esLetra(const char &caracter){
8     if(caracter >= 48 and caracter <= 57) return true;
9     return false;
10 }
11
12 bool esNumero(string::iterator &letra){
13     if(*letra >= 48 and *letra <= 57) return true;
14     return false;
15 }
16
17 bool esLetra(string::iterator &letra){
18     if(*letra >= 97 and *letra <= 122) return true;
19     return false;}
```

2.2. Verificar Identificador con Diagrama de Transiciones

```
1
2 void verificarIdentificador(){
3     char letra;
4     ifstream fe("datos.txt");
```

```

5   int estado = 0;
6   while(!fe.eof()){
7       estado = 0;
8       while(letra != ';' and !fe.eof()){
9           fe>>letra;
10          switch(estado){
11              case 0:
12                  if(letra >= 48 and letra <= 57){
13                      estado = 2;
14                  }
15                  else if((letra >= 97 and letra <= 122) or letra == '._'↵
16                      ){
17                      estado = 1;
18                  }
19                  else{
20                      estado = 2;
21                  }
22                  break;
23              case 1:
24                  if((letra >= 97 and letra <= 122) or (letra >=48 and ↵
25                      letra <= 57) or letra == '._'){
26                      estado = 1;
27                  }
28                  else if(letra == ';'){
29                      estado = 3;
30                  }
31                  else{
32                      estado = 2;
33                  }
34                  break;
35              }
36          if(estado == 3){
37              cout<<"Identificador correcto"<<endl;
38          }
39          else{
40              cout<<"ERROR"<<endl;
41          }
42          fe>>letra;
43      }
44  }

```

2.3. Verificar Identificador con tabla de Transiciones

```

1   bool identificadorConTabla(){
2       string frase;
3       cin>>frase;
4       int entrada;
5       string::iterator letra = frase.begin();
6       int estado = 0;
7       int tabla[3][3] = {{2,1,-1},{-1,-1,-1},{2,2,10}};
8       do{
9           if(esLetra(letra) or *letra == '._'){
10              entrada = 0;
11          }
12          else if(esNumero(letra)){

```

```

13     entrada = 1;
14 }
15 else if(letra == frase.end()){
16     entrada = 2;
17 }
18 else{
19     return false;
20 }
21 estado = tabla[estado][entrada];
22 if(estado == -1)return false;
23 letra++;
24 }while(estado != 10);
25 return true;
26 }

```

2.4. Verificar Número Real con Diagrama de Transiciones

```

1 bool numeroReal(){
2     string frase;
3     cin>>frase;
4     int estado = 1;
5     for(int i = 0; i < frase.size(); i++){
6         switch(estado){
7             case 1:
8                 if(esLetra(frase[i])){
9                     estado = 2;
10                }
11                else{
12                    return false;
13                }
14                break;
15             case 2:
16                 if(esNumero(frase[i])){
17                     estado = 2;
18                }
19                else if(frase[i] == '.'){
20                    estado = 3;
21                }
22                else if(frase[i] == 'E'){
23                    estado = 5;
24                }
25                else{
26                    return false;
27                }
28                break;
29             case 3:
30                 if(esNumero(frase[i])){
31                     estado = 4;
32                }
33                else{
34                    return false;
35                }
36                break;
37             case 4:

```

```

38         if(esNumero(frase[i])){
39             estado = 4;
40         }
41         else if(frase[i] == 'E'){
42             estado = 5;
43         }
44         else{
45             return false;
46         }
47         break;
48     case 5:
49         if(esNumero(frase[i])){
50             estado = 7;
51         }
52         else if(frase[i] == '-' or frase[i] == '+'){
53             estado = 6;
54         }
55         else{
56             return false;
57         }
58         break;
59     case 6:
60         if(esNumero(frase[i])){
61             estado = 7;
62         }
63         else{
64             return false;
65         }
66         break;
67     case 7:
68         if(esNumero(frase[i])){
69             estado = 7;
70         }
71         else{
72             return false;
73         }
74         break;
75     }
76 }
77 if(estado == 4 or estado == 7){
78     return true;
79 }
80 return false;
81 }

```

2.5. Verificar Número Real cond Tabla de Transiciones

```

1 bool numeroRealConTabla(){
2     string frase;
3     cin>>frase;
4     int entrada;
5     string::iterator letra = frase.begin();
6     int estado = 1;
7     int tabla[7][6] = {{2,-1,-1,-1,-1,-1},
8                        {2, 3, 5,-1,-1,-1},
9                        {4,-1,-1,-1,-1,-1},

```

```

10         {4,-1, 5,-1,-1,10},
11         {7,-1,-1, 6, 6,-1},
12         {7,-1,-1,-1,-1,-1},
13         {7,-1,-1,-1,-1,10}};
14     do{
15         if(esNumero(letra))
16             entrada = 0;
17         else if(letra == frase.end())
18             entrada = 5;
19         else{
20             switch(*letra){
21                 case '.': entrada = 1; break;
22                 case 'E': entrada = 2; break;
23                 case '+': entrada = 3; break;
24                 case '-': entrada = 4; break;
25                 default: return false;
26             }
27         }
28         estado = tabla[estado - 1][entrada];
29         letra++;
30         if(estado == -1) return false;
31     } while(estado != 10);
32     return true;
33 }

```

3. Ejemplos

3.1. Verificar Identificador con Diagrama de Transiciones

El archivo "datos.txt" tiene el siguiente contenido.

- temp;
- temp1234;
- 1234temp;
- _ 123;

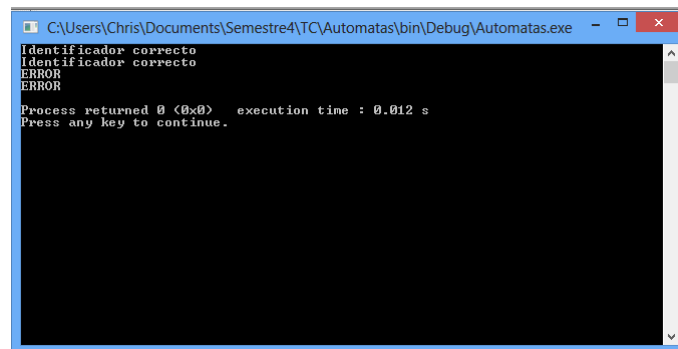


Figura 1: Ejemplo 1

3.2. Verificar Identificador con tabla de Transiciones

Para este ejemplo usaremos los mismos identificadores del archivo "datos.txt" cuyo contenido se mostro anteriormente.

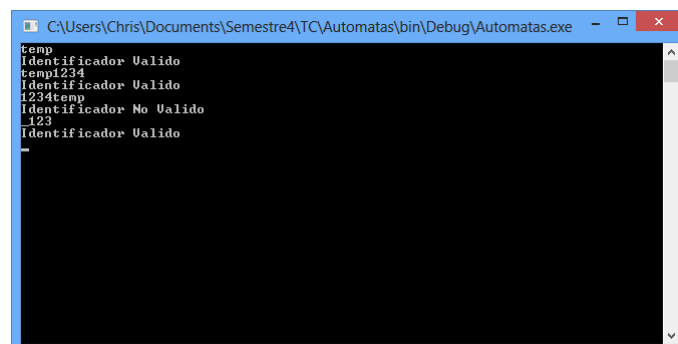
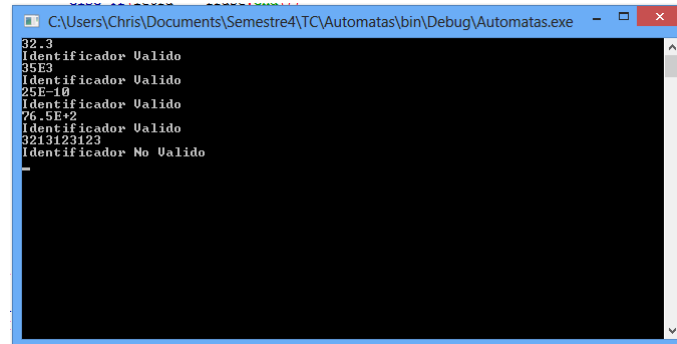


Figura 2: Ejemplo 2

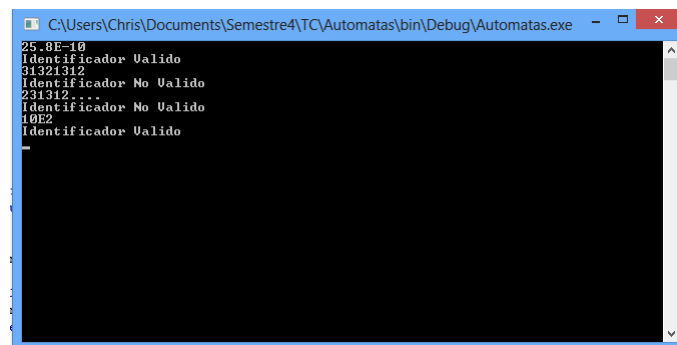
3.3. Verificar Número Real con Diagrama de Transiciones



```
C:\Users\Chris\Documents\Semestre4\TC\Automatas\bin\Debug\Automatas.exe
32.3
Identificador Valido
35E3
Identificador Valido
25E-10
Identificador Valido
76.5E+2
Identificador Valido
3213123123
Identificador No Valido
```

Figura 3: Ejemplo 3

3.4. Verificar Número Real cond Tabla de Transiciones



```
C:\Users\Chris\Documents\Semestre4\TC\Automatas\bin\Debug\Automatas.exe
25.0E-10
Identificador Valido
31321312
Identificador No Valido
231312....
Identificador No Valido
10E2
Identificador Valido
```

Figura 4: Ejemplo 4