

UNIVERSIDAD NACIONAL DE SAN AGUSTIN DE
AREQUIPA



Ciencia de la Computacion

COMPILADORES GRUPO A

YUBER ELMER VELAZCO PAREDES

PRACTICA 02

INTEGRANTES:

PUCHO ZEVALLOS KELVIN PAUL

2020

1. Implemente el algoritmo por diagramas de transición para el reconocimiento de un identificador.

```
#include<iostream>

using namespace std;

int main() {

    char arr[50]="pla45.";

    int simbolo;

    int estado = 1;

    int i=0;
    while (*(arr+i) != '\0')
    {

        for (char j='A',k = 'a'; k <= 'z'; k++,j++)
        {
            if(arr[i] == k || arr[i] == j){
                simbolo = 1;
            }
        }

        for(char l='0';l<='9';l++){
            if(arr[i] == l){
                simbolo = 4;
            }
        }

        switch (estado)
        {
            case 1:
                if(simbolo == 4)
                    estado = 2;
                else if(simbolo == 1)
                    estado = 3;
                break;
            case 2:
                //cout<<"ERROR";
                break;
            case 3:
                if (simbolo == 1)
                    estado = 3;
                else if(simbolo == 4)
                    estado = 3;
                break;
        }
        i++;
    }

    if(estado == 3){
```

```

        cout<<"lectura_completa\n";
    }

    return 0;
}

```

```

judal@judal-OMEN-by-HP-Laptop-15-dc0xxx:~
lectura_completa
judal@judal-OMEN-by-HP-Laptop-15-dc0xxx:~

```

2. Implemente, el algoritmo por tablas de transición para el reconocimiento de un identificador.

```

#include<iostream>
#include<string.h>
using namespace std;

int main() {

    int arr[10][10]={{1,2,10},{10,10,10},{2,2,50}};

    char arreglo[7] = { 'h', 'o', 'l', 'p', 'k', '4', '5' };
    int tam = sizeof(arreglo)/sizeof(arreglo[0]);
    const int size = 6;

    int estado=0;
    int entrada=0;

    int i=0;

    do{

        int simbolo = 1;//para char
        char dato = arreglo[i];
        for(char f='0';f<='9';f++){
            if(dato == f){
                simbolo = 4;//para Numero
            }
        }
        if(tam-1 == i ){
            simbolo = tam-1;
        }

        switch (simbolo)
        {
            case 4:
                entrada = 0;

```

```

        break;
    case 1:
        entrada = 1;
        break;
    case size:
        entrada = 2;
        break;
    default:
        break;
    }
    estado = arr[estado][entrada];

    if(estado == 10){
        break;
    }
    i++;
} while(estado != 50);

if(estado == 50){
    cout<<"lectura completa\n";
}

return 0;
}

```

```

juda1@juda1-OMEN-by-HP-Laptop-15-dc0xxx:~
lectura completa
juda1@juda1-OMEN-by-HP-Laptop-15-dc0xxx:~

```

3. Proponga un diagrama de transición y su correspondiente programa para el reconocimiento de números (Ejemplo: 45,-45, 0.15,-0.15,.67,-.67)

```

#include<iostream>
using namespace std;

int main() {
    char arr[10]="-0.15";

    int simbolo;

    int estado = 1;
    int i=0;
    while (*(arr+i) != '\0')
    {
        for(char l='0';l<='9';l++){

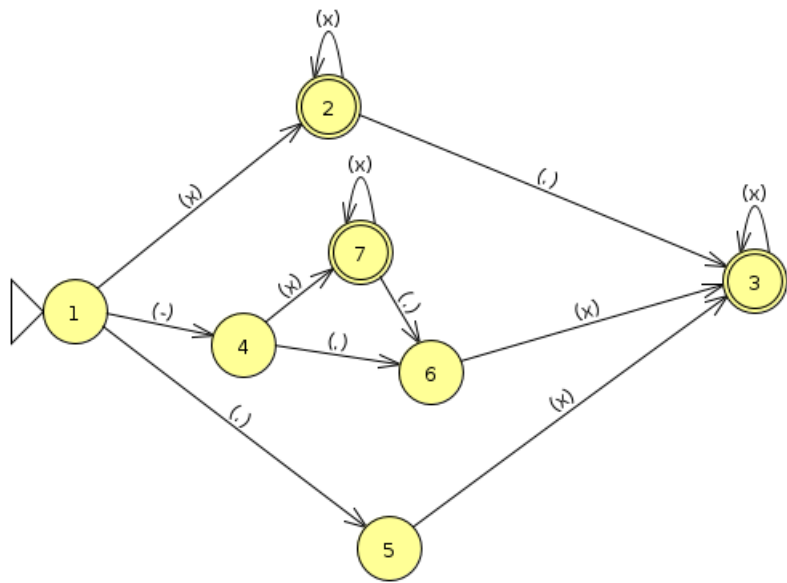
```

```

        if(arr[i] == 1){
            simbolo = 4;
        }
    }
    if(arr[i] == '-') {
        simbolo = 45;
    }
    if(arr[i] == '.') {
        simbolo = 46;
    }
    switch (estado)
    {
    case 1:
        if(simbolo == 4)
            estado = 2;
        else if(simbolo == 45)
            estado = 4;
        else if(simbolo == 46)
            estado = 5;
        break;
    case 2:
        if(simbolo == 4)
            estado = 2;
        else if(simbolo == 45)
            estado = 10;
        else if(simbolo == 46)
            estado = 3;
        break;
    case 3:
        if(simbolo == 4)
            estado = 3;
        else if(simbolo == 45)
            estado = 10;
        else if(simbolo == 46)
            estado = 10;
        break;
    case 4:
        if(simbolo == 4)
            estado = 7;
        else if(simbolo == 45)
            estado = 10;
        else if(simbolo == 46)
            estado = 6;
        break;
    case 5:
        if(simbolo == 4)
            estado = 3;
        else if(simbolo == 45)
            estado = 10;
        else if(simbolo == 46)
            estado = 10;
        break;
    case 6:
        if(simbolo == 4)
            estado = 3;
        else if(simbolo == 45)
            estado = 10;

```

```
        else if(simbolo == 46)
            estado = 10;
        break;
    case 7:
        if(simbolo == 4)
            estado = 7;
        else if(simbolo == 45)
            estado = 10;
        else if(simbolo == 46)
            estado = 6;
        break;
    case 10:
        break;
    }
    i++;
}
if(estado == 2 || estado == 3 || estado == 7){
    cout<<"lectura_completa\n";
}
return 0;
}
```



4. La solución 2, puede ser automatizada, teniendo como dato de entrada la tabla de transiciones. Defina el formato de la tabla de transición y en un archivo de texto plano represente la respectiva tabla (considere el diagrama del ejercicio C). Finalmente, en base al archivo de texto plano, determine la validez de una entrada.

```
#include<iostream>
#include<fstream>

using namespace std;

int main() {

    fstream fichero;

    fichero.open("fichero.txt",ios::in|ios::out);

    string guardar;
    int arr[10][10] = {};

    if (!fichero.fail()){

        int fila=0;
        while (!fichero.eof())
        {

            getline(fichero,guardar);

            int columna = 0;
            //cout<<guardar<<endl;
            for(int i=0 ; guardar[i] != '\0' ; i++){

                if(guardar[i] != ','){
                    int num = guardar[i] - '0';
                    if(guardar[i+1] == '0'){
                        arr[fila][columna] = num*10;
                        cout<<num*10;
                        i++;
                    } else
                    {
                        arr[fila][columna] = num;
                        cout<<num;
                    }
                    columna++;
                } else{
                    cout<<" , ";
                }
            }
        }
    }
```

```

        }
        fila++;
        cout<<endl;
    }
}
else
{
    cout<<"ERROR,no existe dicho archivo\n";
    exit(1);
}
fichero.close();

char* arreglo = new char[5];
string hola;
cout<<"\n_Por_favor ,coloque su digito\n";
getline(cin , hola);
int tam=0;
for (int i = 0; hola[i] != '\0'; i++)
{
    arreglo[i] = hola[i];
    cout<<arreglo[i];
    tam++;
}
tam = tam - 1;

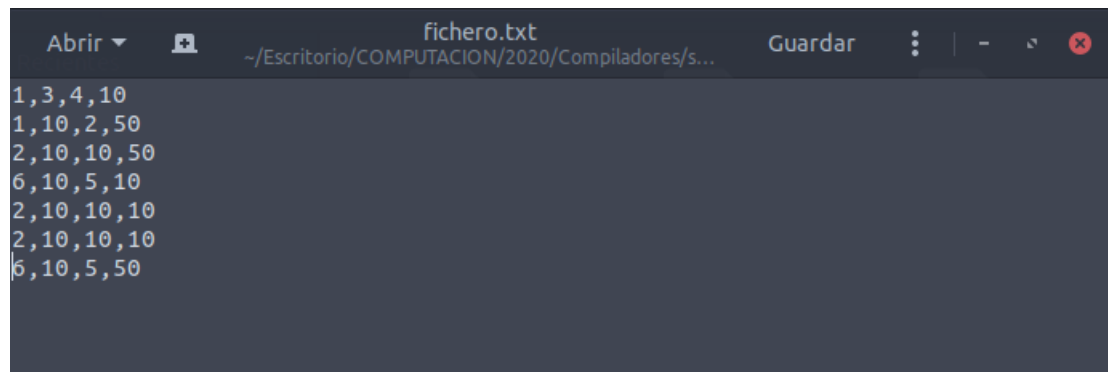
int estado = 0;
int entrada = 0;
string simbolo;
int i=0;
do
{
    for(char l='0';l<='9';l++){
        if(arreglo[i] == l){
            simbolo = "numero";
        }
    }
    if(arreglo[i] == '-')
        simbolo = "guion";
    if(arreglo[i]=='.')
        simbolo = "punto";
    if(tam == i )
        simbolo = "fin";

    //verificacion
    if(simbolo == "numero")
        entrada = 0;
    else if(simbolo == "guion")
        entrada = 1;
    else if(simbolo == "punto")
        entrada = 2;
    else if(simbolo == "fin")
        entrada = 3;
    else
        break;
}

```



```
        estado = arr[estado][entrada];  
        if(estado == 10){  
            break;  
        }  
        i++;  
    } while (estado != 50);  
  
    if(estado == 50){  
        cout<<"\nlectura _completa\n";  
    }  
    return 0;  
}
```



A screenshot of a text editor window titled "fichero.txt" with a dark theme. The window shows a list of numbers arranged in columns. The first column contains the numbers 1, 1, 2, 6, 2, 2, 6. The second column contains 3, 10, 10, 10, 10, 10, 10. The third column contains 4, 2, 10, 5, 10, 10, 5. The fourth column contains 10, 50, 50, 10, 10, 10, 50. The cursor is at the end of the last line.

```
1,3,4,10  
1,10,2,50  
2,10,10,50  
6,10,5,10  
2,10,10,10  
2,10,10,10  
6,10,5,50
```