

Instituto Politécnico Nacional
Escuela Superior de Computo Teoría

Computacional

Programa 03:
Autómata Finito Determinístico que
reconoce todas las palabras reservadas del
lenguaje ANSI C

Alumna: Solis Hernández Viridiana

Fecha de entrega:
31/10/2020

Juárez Martínez Genaro

Introducción

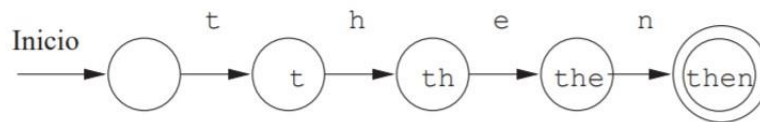
Autómata Finito

Colecciones finitas de estados con reglas de transición que te llevan de un estado a otro.

La mas simple representación de un AF es utilizando un grafo:

- Estados se representan con círculos.
- Estados de transición se representan con arcos entre estados.
- Entradas (lo que causa la transición) se representa con etiquetas sobre los arcos.
- Estados de Aceptación se representan con círculos dobles.

Ejemplo:



Modelo AF para el reconocimiento de la palabra "then".

Autómata Finito Determinista

Un Autómata Finito es "determinista, cuando el autómata no puede encontrarse en más de un estado al mismo tiempo.

Un Autómata Finito Determinista consta de:

1. Un conjunto finito de estados (Q , típicamente).
2. Un alfabeto de entrada (Σ , típicamente).
3. Una función de transición (δ , típicamente).
4. Un estado inicial (q_0 , en Q , típicamente).
5. Un conjunto de estados finales o de aceptación ($F \subseteq Q$)

Palabras Reservadas ANSI C

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

Planteamiento del problema

Se requiere programar el autómata finito determinístico que reconozca todas las palabras reservadas del lenguaje ANSI C.

1. El programa deberá de leer un código (archivo) cualquiera.
2. El autómata deberá de identificar cada palabra reservada, contarlas y al final deberá decir cuantas encontró de cada una de ellas.
3. En un archivo imprimir la evaluación del autómata por cada carácter que lea y cambio de estado, es decir, toda la historia del proceso.
4. En otro archivo enumerar y contar cuantas palabras reservadas fueron encontradas.
5. Tener una opción para ver el autómata, es decir, hay que graficar

Implementación

Al iniciar el programa se solicita el nombre de algún archivo con tu extensión, este se guarda en un arreglo tipo char el cual lo utilizaremos como nombre cuando abramos el archivo.

El autómata preguntara si queremos ver el grafo del autómata.

Iremos leyendo el archivo carácter por carácter, esto nos permitirá ir evaluando en el autómata, pero los únicos caracteres que entraran a ser evaluados serán los que se encuentren en el rango de 97 a 119 que corresponde a su valor en el código ASCII.

Los caracteres se irán evaluando en un switch de 130 case ya que cada case corresponde a cada estado de nuestro autómata.

Si el carácter entra en alguno de los casos del switch se deberá ir guardando en una cadena de caracteres la cual nos servirá al momento de imprimir la historia del autómata.

Utilizamos la librería graphics.h la cual nos ayuda a dibujar el autómata.

Debido a situaciones desconocidas se presentan inconvenientes al momento de utilizar las funciones de graphics.h dentro de un ciclo.

Funcionamiento

Ejecución de "Programa03.cpp", "Programa03.exe"

El programa solicita el nombre del archivo con su extensión.

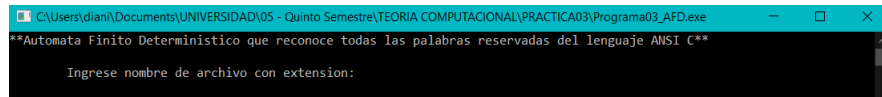
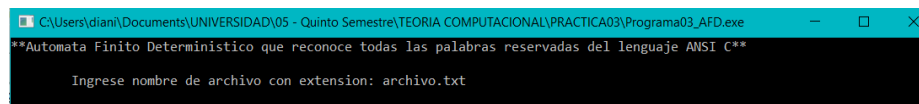


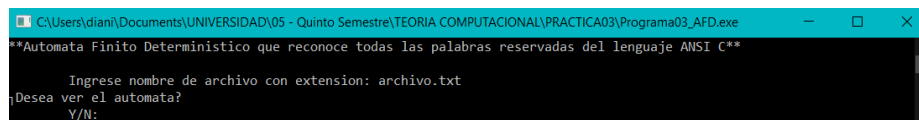
Figura3.

PRUEBA

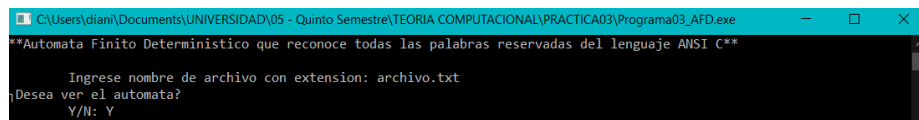
El usuario ingresa nombre del archivo: "archivo.txt"



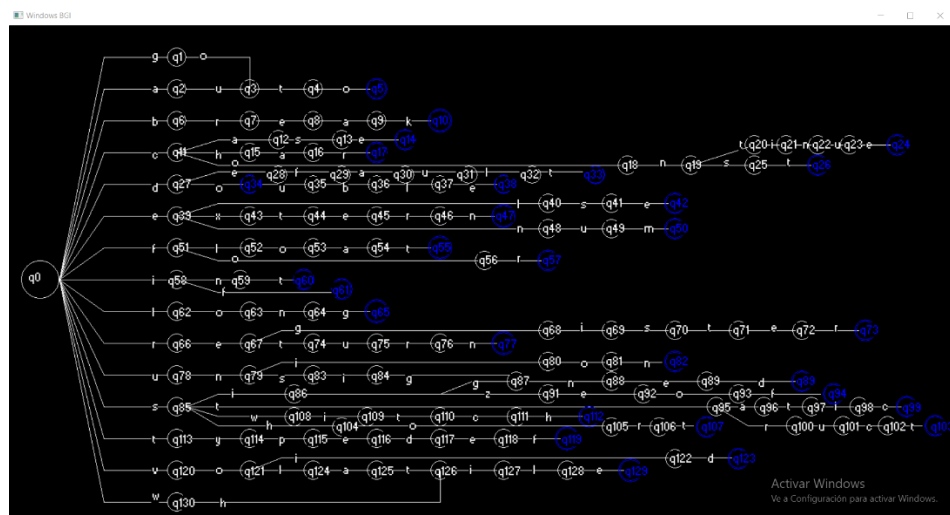
El programa pregunta si se desea ver el autómata.



Se pide ver al autómata.



El autómata muestra el grafo



El programa muestra las palabras reservadas y cuantas veces fueron utilizadas.

```
C:\Users\diani\Documents\UNIVERSIDAD>> EVALUACION TERMINADA <<
Palabras reservadas encontradas:
auto : 1
break : 1
case : 1
char : 1
const : 1
continue : 1
default : 1
do : 1
double : 1
else : 1
enum : 1
extern : 1
float : 1
for : 1
goto : 2
if : 1
int : 1
long : 1
register : 1
return : 1
short : 1
signed : 1
sizeof : 1
static : 1
struct : 1
switch : 1
typedef : 4
union : 4
unsigned : 4
void : 4
volatile : 4
while : 4
-----
```

Archivo Salida 1: “historiaDelProceso.txt”.

```
f(q0 , g )->q1 f(q1 , o )->q3 f(q3 , t )->q4 f(q4 , o )->q5 Acepta cadena: goto
f(q0 , t )->q113 f(q113 , a )-> Rechaza cadena
f(q113 , t )-> Rechaza cadena
f(q113 , i )-> Rechaza cadena
f(q113 , c )-> Rechaza cadena
f(q0 , a )->q2 f(q2 , u )->q3 f(q3 , t )->q4 f(q4 , o )->q5 Acepta cadena: auto
f(q0 , b )->q6 f(q6 , r )->q7 f(q7 , e )->q8 f(q8 , a )->q9 f(q9 , k )->q10 Acepta
cadena: break
f(q0 , g )->q1 f(q1 , o )->q3 f(q3 , o )-> Rechaza cadena
f(q3 , t )->q4 f(q4 , o )->q5 Acepta cadena: goto
f(q0 , c )->q11 f(q11 , a )->q12 f(q12 , s )->q13 f(q13 , e )->q14 Acepta cadena: case
f(q0 , c )->q11 f(q11 , h )->q15 f(q15 , s )-> Rechaza cadena
f(q15 , a )->q16 f(q16 , r )->q17 Acepta cadena: char
f(q0 , c )->q11 f(q11 , o )->q18 f(q18 , n )->q19 f(q19 , s )->q25 f(q25 , t )->q26
Acepta cadena: const
f(q0 , c )->q11 f(q11 , o )->q18 f(q18 , n )->q19 f(q19 , t )->q20 f(q20 , i )->q21
f(q21 , n )->q22 f(q22 , u )->q23 f(q23 , e )->q24 f(q24 , e )-> Rechaza cadena
f(q0 , d )->q27 f(q27 , e )->q28 f(q28 , s )-> Rechaza cadena
f(q28 , f )->q29 f(q29 , a )->q30 f(q30 , u )->q31 f(q31 , l )->q32 f(q32 , t )->q33
f(q33 , a )-> Rechaza cadena
f(q0 , d )->q27 f(q27 , o )->q34 f(q34 , f )-> Rechaza cadena
f(q34 , a )-> Rechaza cadena
f(q0 , d )->q27 f(q27 , o )->q34 f(q34 , u )->q35 f(q35 , b )->q36 f(q36 , l )->q37
f(q37 , e )->q38 Acepta cadena: double
f(q0 , e )->q39 f(q39 , l )->q40 f(q40 , s )->q41 f(q41 , e )->q42 Acepta cadena: else
f(q0 , e )->q39 f(q39 , n )->q48 f(q48 , u )->q49 f(q49 , m )->q50 Acepta cadena: enum
f(q0 , e )->q39 f(q39 , x )->q43 f(q43 , t )->q44 f(q44 , e )->q45 f(q45 , r )->q46
f(q46 , n )->q47 Acepta cadena: extern
f(q0 , f )->q51 f(q51 , l )->q52 f(q52 , o )->q53 f(q53 , a )->q54 f(q54 , t )->q55
Acepta cadena: float
f(q0 , f )->q51 f(q51 , o )->q56 f(q56 , r )->q57 Acepta cadena: for
f(q0 , i )->q58 f(q58 , f )->q61 Acepta cadena: if
f(q0 , i )->q58 f(q58 , n )->q59 f(q59 , t )->q60 Acepta cadena: int
f(q0 , l )->q62 f(q62 , o )->q63 f(q63 , n )->q64 f(q64 , g )->q65 Acepta cadena: long
f(q0 , r )->q66 f(q66 , e )->q67 f(q67 , g )->q68 f(q68 , i )->q69 f(q69 , s )->q70
f(q70 , t )->q71 f(q71 , e )->q72 f(q72 , r )->q73 Acepta cadena: register
f(q0 , r )->q66 f(q66 , e )->q67 f(q67 , t )->q74 f(q74 , u )->q75 f(q75 , r )->q76
f(q76 , n )->q77 Acepta cadena: return
```

Archivo Salida 2: “contadorDePalabras.txt”.

Muestra las palabras enumeradas y las veces que se repitieron.

```
1. auto 1
2. break 1
3. case 1
4. char 1
5. const 1
6. continue 1
7. default 1
8. do 1
9. double 1
10. else 1
11. enum 1
12. extern 1
13. float 1
14. for 1
15. goto 2
16. if 1
17. int 1
18. long 1
19. register 1
20. return 1
21. short 1
22. signed 1
23. sizeof 1
24. static 1
25. struct 1
26. switch 1
27. typedef 4
28. union 4
29. unsigned 4
30. void 4
31. volatile 4
32. while 4
```

Conclusión

Debido a que el autómata era determinista fue fácil de implementar gracias a que la mayoría del código en los estados tiene comportamiento similar y siempre se tiene que ir a un estado diferente.

La parte complicada de esta practica se llevo a cabo al momento de graficar debido a que si se utilizaban las funciones de la librería graphics.h dentro de ciclos algo parecía fallar.

Anexo

```
#include<iostream>
#include<stdlib.h>
#include<string.h>
#include<fstream>
#include<graphics.h>
#include<math.h>
#include<cstring>

// 97 - 119 ->Letras minúsculas en ASCII
//Variables Globales
FILE *archivo,*evaluacion,*enumeracion,*contar;
char caracter;
int estado=0;
char unionCadenas[] = "";
int x=1500, y=800;
int radio = 30;
int centrox = 50;
int centroy = y/2;
int separacion =50;
int time = 100;
char acepta = 'F';
char nombreArchivo[] = "";
int palabras=0;
int contador[31] = {0};
//Prototipo de funciones
void Menu();
void abrirArchivo();
void Estados();
void archivoHistoria();
void mostrarAutomata();
void archivoContador();
void contadorPalabras();
void imprimirContador();

using namespace std;
int main(){
    cout<<"**Automata Finito Deterministico que reconoce todas las palabras reservadas del lenguaje ANSI
C**"<<endl;
    Menu();
    getch();
}

void Menu(){
    char opcion;
    cout<<"\n\tIngrese nombre de archivo con extension: ";
    cin>>nombreArchivo;

    cout<<"¿Desea ver el automata?"<<endl;
    cout<<"\tY/N: ";
    cin>>opcion;
    if ( opcion == 89 ){
        //cout<<"\nDibujando automata ..... "<<endl;
        system("cls");
        mostrarAutomata();

    }
    else if ( opcion == 78){
        initwindow(x/2,y/4);
        settextstyle(2,0,8);
        outtextxy(50,50,"**EVALUANDO ARCHIVO DE TEXTO**");
        delay(2500);
        abrirArchivo();
        outtextxy(50,50,"**ARCHIVO DE TEXTO EVALUADO**");
        delay(2500);
    }
}
```

```

    }
    exit(1);
}

void abrirArchivo(){

    char numero[]="";
    archivo = fopen(nombreArchivo, "r"); //Abrir archivo para ir leyendo
    archivoHistoria();
    archivoContador();
    palabras = 0;
    if(archivo == NULL)
    {
        cout<<"ERROR al abrir archivo"<<endl;
        exit(1);
    }
    do
    {
        caracter = fgetc(archivo);

        if(!(caracter > 96 && caracter<123)){ //- si esto se cumple ir al estado principal

            if (acepta == 'T'){
                fprintf(evaluacion, " Acepta cadena: %s\n", unionCadenas);
                palabras = palabras + 1;
                acepta = 'F';
            }
            contadorPalabras();
            estado = 0;
            unionCadenas[0] = 0;
        }
        else{

            fprintf(evaluacion, " f(q%d , %c )->", estado, caracter);
            Estados();
        }
    }while(caracter != EOF);

    fclose(archivo);
    fclose(evaluacion); //Cerrando archivo
    cout<<">> EVALUACION TERMINADA <<"<<endl;
    imprimirContador();
    fclose(contar);
}

void archivoHistoria(){
    evaluacion = fopen("historiaDelProceso.txt", "w"); //Abrir archivo para escribir en el

    if(evaluacion == NULL)
    {
        cout<<"ERROR al abrir archivo"<<endl;
        exit(1);
    }
}

void archivoContador(){
    contar = fopen("contadorPalabras.txt", "w"); //Abrir archivo para escribir en el

    if(contar == NULL)
    {
        cout<<"ERROR al abrir archivo"<<endl;
        exit(1);
    }
}

```



```

    }
}

void contadorPalabras(){

    if( strcmp("auto",unionCadenas) == 0 )
    {
        contador[0] += 1;
    }
    else if( strcmp(unionCadenas,"break") == 0 )
    {
        contador[1] += 1;
    }
    else if( strcmp(unionCadenas,"case") == 0 )
    {
        contador[2] += 1;
    }
    else if( strcmp(unionCadenas,"char") == 0 )
    {
        contador[3] += 1;
    }
    else if( strcmp(unionCadenas,"const") == 0 )
    {
        contador[4] += 1;
    }
    else if( strcmp(unionCadenas,"continue") == 0 )
    {
        contador[5] += 1;
    }
    else if( strcmp(unionCadenas,"default") == 0 )
    {
        contador[6] += 1;
    }
    else if( strcmp(unionCadenas,"do") == 0 )
    {
        contador[7] += 1;
    }
    else if( strcmp(unionCadenas,"double") == 0 )
    {
        contador[8] += 1;
    }
    else if( strcmp(unionCadenas,"else") == 0 )
    {
        contador[9] += 1;
    }
    else if( strcmp(unionCadenas,"enum") == 0 )
    {
        contador[10] += 1;
    }
    else if( strcmp(unionCadenas,"extern") == 0 )
    {
        contador[11] += 1;
    }
    else if( strcmp(unionCadenas,"float") == 0 )
    {
        contador[12] += 1;
    }
    else if( strcmp(unionCadenas,"for") == 0 )
    {
        contador[13] += 1;
    }
    else if( strcmp(unionCadenas,"goto") == 0 )
    {
        contador[14] += 1;
    }
    else if( strcmp(unionCadenas,"if") == 0 )
    {

```

```

        contador[15] += 1;
    }
    else if( strcmp(unionCadenas,"int") == 0 )
    {
        contador[16] += 1;
    }
    else if( strcmp(unionCadenas,"long") == 0 )
    {
        contador[17] += 1;
    }
    else if( strcmp(unionCadenas,"register") == 0 )
    {
        contador[18] += 1;
    }
    else if( strcmp(unionCadenas,"return") == 0 )
    {
        contador[19] += 1;
    }
    else if( strcmp(unionCadenas,"short") == 0 )
    {
        contador[20] += 1;
    }
    else if( strcmp(unionCadenas,"signed") == 0 )
    {
        contador[21] += 1;
    }
    else if( strcmp(unionCadenas,"sizeof") == 0 )
    {
        contador[22] += 1;
    }
    else if( strcmp(unionCadenas,"static") == 0 )
    {
        contador[23] += 1;
    }
    else if( strcmp(unionCadenas,"struct") == 0 )
    {
        contador[24] += 1;
    }
    else if( strcmp(unionCadenas,"switch") == 0 )
    {
        contador[25] += 1;
    }
    else if( strcmp(unionCadenas,"typedef") == 0 )
    {
        contador[26] += 1;
    }
    else if( strcmp(unionCadenas,"union") == 0 )
    {
        contador[27] += 1;
    }
    else if( strcmp(unionCadenas,"unsigned") == 0 )
    {
        contador[28] += 1;
    }
    else if( strcmp(unionCadenas,"void") == 0 )
    {
        contador[29] += 1;
    }
    else if( strcmp(unionCadenas,"volatile") == 0 )
    {
        contador[30] += 1;
    }
    else if( strcmp(unionCadenas,"while") == 0 )
    {
        contador[31] += 1;
    }
}

```

```

}

void imprimirContador(){
    cout<<"Palabras reservadas encontradas:"<<endl;

    cout<<"auto : "<<contador[0]<<endl;
    cout<<"break : "<<contador[1]<<endl;
    cout<<"case : "<<contador[2]<<endl;
    cout<<"char : "<<contador[3]<<endl;
    cout<<"const : "<<contador[4]<<endl;
    cout<<"continue : "<<contador[5]<<endl;
    cout<<"default : "<<contador[6]<<endl;
    cout<<"do : "<<contador[7]<<endl;
    cout<<"double : "<<contador[8]<<endl;
    cout<<"else : "<<contador[9]<<endl;
    cout<<"enum : "<<contador[10]<<endl;
    cout<<"extern : "<<contador[11]<<endl;
    cout<<"float : "<<contador[12]<<endl;
    cout<<"for : "<<contador[13]<<endl;
    cout<<"goto : "<<contador[14]<<endl;
    cout<<"if : "<<contador[15]<<endl;
    cout<<"int : "<<contador[16]<<endl;
    cout<<"long : "<<contador[17]<<endl;
    cout<<"register : "<<contador[18]<<endl;
    cout<<"return : "<<contador[19]<<endl;
    cout<<"short : "<<contador[20]<<endl;
    cout<<"signed : "<<contador[21]<<endl;
    cout<<"sizeof : "<<contador[22]<<endl;
    cout<<"static : "<<contador[23]<<endl;
    cout<<"struct : "<<contador[24]<<endl;
    cout<<"switch : "<<contador[25]<<endl;
    cout<<"typedef : "<<contador[26]<<endl;
    cout<<"union : "<<contador[27]<<endl;
    cout<<"unsigned : "<<contador[28]<<endl;
    cout<<"void : "<<contador[29]<<endl;
    cout<<"volatile : "<<contador[30]<<endl;
    cout<<"while : "<<contador[31]<<endl;

    fprintf(contar,"1. auto %d\n",contador[0]);
    fprintf(contar,"2. break %d\n",contador[1]);
    fprintf(contar,"3. case %d\n",contador[2]);
    fprintf(contar,"4. char %d\n",contador[3]);
    fprintf(contar,"5. const %d\n",contador[4]);
    fprintf(contar,"6. continue %d\n",contador[5]);
    fprintf(contar,"7. default %d\n",contador[6]);
    fprintf(contar,"8. do %d\n",contador[7]);
    fprintf(contar,"9. double %d\n",contador[8]);
    fprintf(contar,"10. else %d\n",contador[9]);
    fprintf(contar,"11. enum %d\n",contador[10]);
    fprintf(contar,"12. extern %d\n",contador[11]);
    fprintf(contar,"13. float %d\n",contador[12]);
    fprintf(contar,"14. for %d\n",contador[13]);
    fprintf(contar,"15. goto %d\n",contador[14]);
    fprintf(contar,"16. if %d\n",contador[15]);
    fprintf(contar,"17. int %d\n",contador[16]);
    fprintf(contar,"18. long %d\n",contador[17]);
    fprintf(contar,"19. register %d\n",contador[18]);
    fprintf(contar,"20. return %d\n",contador[19]);
    fprintf(contar,"21. short %d\n",contador[20]);
    fprintf(contar,"22. signed %d\n",contador[21]);
    fprintf(contar,"23. sizeof %d\n",contador[22]);
    fprintf(contar,"24. static %d\n",contador[23]);
    fprintf(contar,"25. struct %d\n",contador[24]);
    fprintf(contar,"26. switch %d\n",contador[25]);
    fprintf(contar,"27. typedef %d\n",contador[26]);
    fprintf(contar,"28. union %d\n",contador[27]);
    fprintf(contar,"29. unsigned %d\n",contador[28]);

```

```

    fprintf(contar,"30. void %d\n",contador[29]);
    fprintf(contar,"31. volatile %d\n",contador[30]);
    fprintf(contar,"32. while %d\n",contador[31]);
}

void Estados(){

    switch(estado){
        case 0:

            if(caracter == 'g'){
                estado = 1;
                strcat(unionCadenas,"g");
                fprintf(evaluacion,"q%d",estado);
                //cout<<"\ncharacter: "<<caracter<<" estado: "<<estado;
            }
            else if(caracter == 'a')
            {
                estado = 2;
                strcat(unionCadenas,"a");
                fprintf(evaluacion,"q%d",estado);
                //cout<<"\ncharacter: "<<caracter<<" estado: "<<estado;
            }
            else if(caracter == 'b')
            {
                estado = 6;
                strcat(unionCadenas,"b");
                fprintf(evaluacion,"q%d",estado);
                //cout<<"\ncharacter: "<<caracter<<" estado: "<<estado;
            }
            else if(caracter == 'c')
            {
                estado = 11;
                strcat(unionCadenas,"c");
                fprintf(evaluacion,"q%d",estado);
                //cout<<"\ncharacter: "<<caracter<<" estado: "<<estado;
            }
            else if(caracter == 'd')
            {
                estado = 27;
                strcat(unionCadenas,"d");
                fprintf(evaluacion,"q%d",estado);
                //cout<<"\ncharacter: "<<caracter<<" estado: "<<estado;
            }
            else if(caracter == 'e')
            {
                estado = 39;
                strcat(unionCadenas,"e");
                fprintf(evaluacion,"q%d",estado);
                //cout<<"\ncharacter: "<<caracter<<" estado: "<<estado;
            }
            else if(caracter == 'f')
            {
                estado = 51;
                strcat(unionCadenas,"f");
                fprintf(evaluacion,"q%d",estado);
            }
            else if(caracter == 'i')
            {
                estado = 58;
                strcat(unionCadenas,"i");
                fprintf(evaluacion,"q%d",estado);
            }
            else if(caracter == 'l')
            {
                estado = 62;
            }
        }
    }
}

```

```

        strcat(unionCadenas,"l");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'r')
    {
        estado = 66;
        strcat(unionCadenas,"r");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'u')
    {
        estado = 78;
        strcat(unionCadenas,"u");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 's')
    {
        estado = 85;
        strcat(unionCadenas,"s");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 't')
    {
        estado = 113;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'v')
    {
        estado = 120;
        strcat(unionCadenas,"v");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'w')
    {
        estado = 130;
        strcat(unionCadenas,"w");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 1:

    if(caracter == 'o'){
        estado = 3;
        strcat(unionCadenas,"o");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 2:

    if(caracter == 'u'){
        estado = 3;
        strcat(unionCadenas,"u");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    }
}

```

```

break;
case 3:
    if(caracter == 't'){
        estado = 4;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 4:
    if(caracter == 'o'){
        estado = 5;
        strcat(unionCadenas,"o");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    }

break;

case 5:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);

break;

case 6:
    if(caracter == 'r'){
        estado = 7;
        strcat(unionCadenas,"r");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 7:
    if(caracter == 'e'){
        estado = 8;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 8:
    if(caracter == 'a'){
        estado = 9;
        strcat(unionCadenas,"a");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 9:
    if(caracter == 'k'){

```

```

                                estado = 10;
                                strcat(unionCadenas,"k");
                                fprintf(evaluacion,"q%d",estado);
                                acepta = 'T';
                                //cout<<"cadena: "<<unionCadenas<<endl;
                                }
                                else{
                                        acepta = 'F';
                                        fputs(" Rechaza cadena\n",evaluacion);
                                }
break;
case 10:
                                acepta = 'F';
                                fputs(" Rechaza cadena\n",evaluacion);
break;
case 11:
                                if(caracter == 'a'){
                                        estado = 12;
                                        strcat(unionCadenas,"a");
                                        fprintf(evaluacion,"q%d",estado);
                                }
                                else if(caracter == 'h')
                                {
                                        estado = 15;
                                        strcat(unionCadenas,"h");
                                        fprintf(evaluacion,"q%d",estado);
                                }
                                else if(caracter == 'o')
                                {
                                        estado = 18;
                                        strcat(unionCadenas,"o");
                                        fprintf(evaluacion,"q%d",estado);
                                }
                                else{
                                        acepta = 'F';
                                        estado =131;fputs(" Rechaza cadena\n",evaluacion);
                                }
break;
case 12:
                                if(caracter == 's')
                                {
                                        estado = 13;
                                        strcat(unionCadenas,"s");
                                        fprintf(evaluacion,"q%d",estado);
                                }
                                else{
                                        acepta = 'F';

                                        fputs(" Rechaza cadena\n",evaluacion);
                                }
break;
case 13:
                                if(caracter == 'e'){
                                        estado = 14;
                                        strcat(unionCadenas,"e");
                                        fprintf(evaluacion,"q%d",estado);
                                        acepta = 'T';
                                        //cout<<"cadena: "<<unionCadenas<<endl;
                                }
                                else{
                                        acepta = 'F';

                                        fputs(" Rechaza cadena\n",evaluacion);
                                }
break;
case 14:
                                acepta = 'F';

```

```

                                fputs(" Rechaza cadena\n",evaluacion);
break;
case 15:
    if(caracter == 'a'){
        estado = 16;
        strcat(unionCadenas,"a");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 16:
    if(caracter == 'r'){
        estado = 17;
        strcat(unionCadenas,"r");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 17:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
break;
case 18:
    if(caracter == 'n'){
        estado = 19;
        strcat(unionCadenas,"n");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 19:
    if(caracter == 't'){
        estado = 20;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 's'){
        estado = 25;
        strcat(unionCadenas,"s");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 20:
    if(caracter == 'i'){
        estado = 21;
        strcat(unionCadenas,"i");
        fprintf(evaluacion,"q%d",estado);
    }

```



```

        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);

        }
break;
case 21:
    if(caracter == 'n'){
        estado = 22;
        strcat(unionCadenas,"n");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);

    }
break;
case 22:
    if(caracter == 'u'){
        estado = 23;
        strcat(unionCadenas,"u");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);

    }
break;
case 23:
    if(caracter == 'e'){
        estado = 24;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);

    }
break;
case 24:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 25:
    if(caracter == 't'){
        estado = 26;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);

    }
break;
case 26:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 27:

```

```

        if(caracter == 'e'){
            estado = 28;
            strcat(unionCadenas,"e");
            fprintf(evaluacion,"q%d",estado);
        }
        /*****/
        else if(caracter == 'o'){
            estado = 34;
            strcat(unionCadenas,"o");
            fprintf(evaluacion,"q%d",estado);
            acepta = 'T';
            //cout<<"cadena: "<<unionCadenas<<endl;
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
break;
case 28:
        if(caracter == 'f'){
            estado = 29;
            strcat(unionCadenas,"f");
            fprintf(evaluacion,"q%d",estado);
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
break;
case 29:
        if(caracter == 'a'){
            estado = 30;
            strcat(unionCadenas,"a");
            fprintf(evaluacion,"q%d",estado);
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
break;
case 30:
        if(caracter == 'u'){
            estado = 31;
            strcat(unionCadenas,"u");
            fprintf(evaluacion,"q%d",estado);
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
break;
case 31:
        if(caracter == 'l'){
            estado = 32;
            strcat(unionCadenas,"l");
            fprintf(evaluacion,"q%d",estado);
        }
        else{
            acepta = 'F';

```

```

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 32:
    if(caracter == 't'){
        estado = 33;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 33:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 34:
    if(caracter == 'u'){
        estado = 35;
        strcat(unionCadenas,"u");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 35:
    if(caracter == 'b'){
        estado = 36;
        strcat(unionCadenas,"b");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 36:
    if(caracter == 'l'){
        estado = 37;
        strcat(unionCadenas,"l");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 37:
    if(caracter == 'e'){
        estado = 38;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

```

```

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 38:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
break;
case 39:
    if(caracter == 'l'){
        estado = 40;
        strcat(unionCadenas,"l");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'x'){
        estado = 43;
        strcat(unionCadenas,"x");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'n'){
        estado = 48;
        strcat(unionCadenas,"n");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 40:
    if(caracter == 's'){
        estado = 41;
        strcat(unionCadenas,"s");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 41:
    if(caracter == 'e'){
        estado = 42;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 42:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
break;
case 43:
    if(caracter == 't'){
        estado = 44;
        strcat(unionCadenas,"t");
    }

```

```

        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 44:
    if(caracter == 'e'){
        estado = 45;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 45:
    if(caracter == 'r'){
        estado = 46;
        strcat(unionCadenas,"r");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 46:
    if(caracter == 'n'){
        estado = 47;
        strcat(unionCadenas,"n");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 47:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 48:
    if(caracter == 'u'){
        estado = 49;
        strcat(unionCadenas,"u");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 49:
    if(caracter == 'm'){
        estado = 50;

```

```

        strcat(unionCadenas,"m");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 50:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
break;
case 51:
    if(caracter == 'l'){
        estado = 52;
        strcat(unionCadenas,"l");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'o'){
        estado = 56;
        strcat(unionCadenas,"o");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 52:
    if(caracter == 'o'){
        estado = 53;
        strcat(unionCadenas,"o");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 53:
    if(caracter == 'a'){
        estado = 54;
        strcat(unionCadenas,"a");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 54:
    if(caracter == 't'){
        estado = 55;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';
    }

```

```

        fputs(" Rechaza cadena\n",evaluacion);
    }
    break;
case 55:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    break;
case 56:
        if(caracter == 'r'){
            estado = 57;
            strcat(unionCadenas,"r");
            fprintf(evaluacion,"q%d",estado);
            acepta = 'T';
            //cout<<"cadena: "<<unionCadenas<<endl;
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
    break;
case 57:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    break;
case 58:
        if(caracter == 'f'){
            estado = 61;
            strcat(unionCadenas,"f");
            fprintf(evaluacion,"q%d",estado);
            acepta = 'T';
            //cout<<"cadena: "<<unionCadenas<<endl;
        }
        else if(caracter == 'n'){
            estado = 59;
            strcat(unionCadenas,"n");
            fprintf(evaluacion,"q%d",estado);
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
    break;
case 59:
        if(caracter == 't'){
            estado = 60;
            strcat(unionCadenas,"t");
            fprintf(evaluacion,"q%d",estado);
            acepta = 'T';
            //cout<<"cadena: "<<unionCadenas<<endl;
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
    break;
case 60:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    break;
case 61:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    break;
case 62:

```

```

        if(caracter == 'o'){
            estado = 63;
            strcat(unionCadenas,"o");
            fprintf(evaluacion,"q%d",estado);
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
break;
case 63:
        if(caracter == 'n'){
            estado = 64;
            strcat(unionCadenas,"n");
            fprintf(evaluacion,"q%d",estado);
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
break;
case 64:
        if(caracter == 'g'){
            estado = 65;
            strcat(unionCadenas,"g");
            fprintf(evaluacion,"q%d",estado);
            acepta = 'T';
            //cout<<"cadena: "<<unionCadenas<<endl;
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
break;
case 65:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
break;
case 66:
        if(caracter == 'e'){
            estado = 67;
            strcat(unionCadenas,"e");
            fprintf(evaluacion,"q%d",estado);
        }
        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);
        }
break;
case 67:
        if(caracter == 'g'){
            estado = 68;
            strcat(unionCadenas,"g");
            fprintf(evaluacion,"q%d",estado);
        }
        else if(caracter == 't'){
            estado = 74;
            strcat(unionCadenas,"t");
            fprintf(evaluacion,"q%d",estado);
        }
        else{
            acepta = 'F';

```



```

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 68:
    if(caracter == 'i'){
        estado = 69;
        strcat(unionCadenas,"i");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 69:
    if(caracter == 's'){
        estado = 70;
        strcat(unionCadenas,"s");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 70:
    if(caracter == 't'){
        estado = 71;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 71:
    if(caracter == 'e'){
        estado = 72;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 72:
    if(caracter == 'r'){
        estado = 73;
        strcat(unionCadenas,"r");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 73:
    acepta = 'F';

```

```

                                fputs(" Rechaza cadena\n",evaluacion);
break;
case 74:
    if(caracter == 'u'){
        estado = 75;
        strcat(unionCadenas,"u");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 75:
    if(caracter == 'r'){
        estado = 76;
        strcat(unionCadenas,"r");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 76:
    if(caracter == 'n'){
        estado = 77;
        strcat(unionCadenas,"n");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 77:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);

break;
case 78:
    if(caracter == 'n'){
        estado = 79;
        strcat(unionCadenas,"n");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }

break;
case 79:
    if(caracter == 'i'){
        estado = 80;
        strcat(unionCadenas,"i");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 's'){
        estado = 83;
        strcat(unionCadenas,"s");
        fprintf(evaluacion,"q%d",estado);
    }
}

```

```

        else{
            acepta = 'F';

            fputs(" Rechaza cadena\n",evaluacion);

        }
break;
case 80:
    if(caracter == 'o'){
        estado = 81;
        strcat(unionCadenas,"o");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);

    }
break;
case 81:
    if(caracter == 'n'){
        estado = 82;
        strcat(unionCadenas,"n");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);

    }
break;
case 82:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 83:
    if(caracter == 'i'){
        estado = 84;
        strcat(unionCadenas,"i");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);

    }
break;
case 84:
    if(caracter == 'g'){
        estado = 87;
        strcat(unionCadenas,"g");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);

    }
break;
case 85:
    if(caracter == 'i'){
        estado = 86;
        strcat(unionCadenas,"i");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 't'){

```

```

        estado = 95;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'h'){
        estado = 104;
        strcat(unionCadenas,"h");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'w'){
        estado = 108;
        strcat(unionCadenas,"w");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 86:
    if(caracter == 'g'){
        estado = 87;
        strcat(unionCadenas,"g");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'z'){
        estado = 91;
        strcat(unionCadenas,"z");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 87:
    if(caracter == 'n'){
        estado = 88;
        strcat(unionCadenas,"n");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 88:
    if(caracter == 'e'){
        estado = 89;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 89:
    if(caracter == 'd'){
        estado = 90;
        strcat(unionCadenas,"d");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
    }

```

```

//cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 90:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
break;
case 91:
    if(caracter == 'e'){
        estado = 92;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 92:
    if(caracter == 'o'){
        estado = 93;
        strcat(unionCadenas,"o");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 93:
    if(caracter == 'f'){
        estado = 94;
        strcat(unionCadenas,"f");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 94:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
break;
case 95:
    if(caracter == 'a'){
        estado = 96;
        strcat(unionCadenas,"a");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'r'){
        estado = 100;
        strcat(unionCadenas,"r");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

```

```

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 96:
    if(caracter == 't'){
        estado = 97;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 97:
    if(caracter == 'i'){
        estado = 98;
        strcat(unionCadenas,"i");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 98:
    if(caracter == 'c'){
        estado = 99;
        strcat(unionCadenas,"c");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 99:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 100:
    if(caracter == 'u'){
        estado = 101;
        strcat(unionCadenas,"u");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 101:
    if(caracter == 'c'){
        estado = 102;
        strcat(unionCadenas,"c");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

```

```

        fputs(" Rechaza cadena\n",evaluacion);
    }
    break;
case 102:
    if(caracter == 't'){
        estado = 103;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
    break;
case 103:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    break;
case 104:
    if(caracter == 'o'){
        estado = 105;
        strcat(unionCadenas,"o");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
    break;
case 105:
    if(caracter == 'r'){
        estado = 106;
        strcat(unionCadenas,"r");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
    break;
case 106:
    if(caracter == 't'){
        estado = 107;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
    break;
case 107:
        acepta = 'F';
        fputs(" Rechaza cadena\n",evaluacion);
    break;
case 108:
    if(caracter == 'i'){
        estado = 109;
        strcat(unionCadenas,"i");

```

```

        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 109:
    if(caracter == 't'){
        estado = 110;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 110:
    if(caracter == 'c'){
        estado = 111;
        strcat(unionCadenas,"c");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 111:
    if(caracter == 'h'){
        estado = 112;
        strcat(unionCadenas,"h");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 112:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 113:
    if(caracter == 'y'){
        estado = 114;
        strcat(unionCadenas,"y");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 114:
    if(caracter == 'p'){
        estado = 115;
        strcat(unionCadenas,"p");
        fprintf(evaluacion,"q%d",estado);
    }

```



```

    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 115:
    if(caracter == 'e'){
        estado = 116;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 116:
    if(caracter == 'd'){
        estado = 117;
        strcat(unionCadenas,"d");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 117:
    if(caracter == 'e'){
        estado = 118;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 118:
    if(caracter == 'f'){
        estado = 119;
        strcat(unionCadenas,"f");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 119:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 120:
    if(caracter == 'o'){
        estado = 121;
        strcat(unionCadenas,"o");
        fprintf(evaluacion,"q%d",estado);
    }
}

```

```

else{
    acepta = 'F';

    fputs(" Rechaza cadena\n",evaluacion);
}
break;
case 121:
    if(caracter == 'i'){
        estado = 122;
        strcat(unionCadenas,"i");
        fprintf(evaluacion,"q%d",estado);
    }
    else if(caracter == 'l'){
        estado = 124;
        strcat(unionCadenas,"l");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 122:
    if(caracter == 'd'){
        estado = 123;
        strcat(unionCadenas,"d");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 123:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 124:
    if(caracter == 'a'){
        estado = 125;
        strcat(unionCadenas,"a");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 125:
    if(caracter == 't'){
        estado = 126;
        strcat(unionCadenas,"t");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 126:
    if(caracter == 'i'){

```

```

        estado = 127;
        strcat(unionCadenas,"i");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 127:
    if(caracter == 'l'){
        estado = 128;
        strcat(unionCadenas,"l");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 128:
    if(caracter == 'e'){
        estado = 129;
        strcat(unionCadenas,"e");
        fprintf(evaluacion,"q%d",estado);
        acepta = 'T';
        //cout<<"cadena: "<<unionCadenas<<endl;
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
case 129:
    acepta = 'F';
    fputs(" Rechaza cadena\n",evaluacion);
break;
case 130:
    if(caracter == 'h'){
        estado = 126;
        strcat(unionCadenas,"h");
        fprintf(evaluacion,"q%d",estado);
    }
    else{
        acepta = 'F';

        fputs(" Rechaza cadena\n",evaluacion);
    }
break;
    }
}

void mostrarAutomata(){
    initwindow(x,y);
    setttextstyle(2,0,7);
    //Dibujamos estado q0;
    circle(centrox,centroy,radio);
    outtextxy(radio,centroy-(radio/2),"q0");
    for(int i=1; i<=7; i++){
        line(centrox+radio,centroy,centrox*3,centroy-(i*separacion));
        delay(time);
        line(centrox*3,centroy-i*separacion,centrox*5,centroy-i*separacion);
    }
}

```

```

delay(time);
if(i==7)
{
    //goto
    line(centrox*5+radio,centroy-
i*separacion,centrox*7+radio,centroy-i*separacion);
    outtextxy(centrox*4.5,centroy-i*(separacion+2),"g");
    circle(centrox*5+(radio/2),centroy-i*separacion,radio/2);
    outtextxy(centrox*5+(radio/4),centroy-12-i*separacion,"q1");

    outtextxy(centrox*6,centroy-i*(separacion+2),"o");
    line(centrox*7+radio,centroy-
i*separacion,centrox*7+radio,centroy-6*separacion);
    delay(time);
}
else if(i==6)
{
    //auto
    line(centrox*5+radio,centroy-
i*separacion,centrox*11+radio,centroy-i*separacion);
    outtextxy(centrox*4.5,centroy-i*(separacion+2),"a");
    circle(centrox*5+(radio/2),centroy-i*separacion,radio/2);
    outtextxy(centrox*5+(radio/4),centroy-12-i*separacion,"q2");

    outtextxy(centrox*6.5,centroy-i*(separacion+2),"u");
    circle(centrox*7+radio,centroy-i*separacion,radio/2);
    outtextxy(centrox*7+20,centroy-12-i*separacion,"q3");

    outtextxy(centrox*8.5,centroy-i*(separacion+2),"t");
    circle(centrox*9+radio,centroy-i*separacion,radio/2);
    outtextxy(centrox*9+20,centroy-12-i*separacion,"q4");

    outtextxy(centrox*10.5,centroy-i*(separacion+2),"o");
    setcolor(BLUE);
    circle(centrox*11+radio,centroy-i*separacion,radio/2);
    circle(centrox*11+radio,centroy-i*separacion,(radio/2)+2);
    outtextxy(centrox*11+20,centroy-12-i*separacion,"q5");
    setcolor(WHITE);
    delay(time);
}
else if(i==5)
{
    //break
    line(centrox*5+radio,centroy-
i*separacion,centrox*13+radio,centroy-i*separacion);
    outtextxy(centrox*4.5,centroy-i*(separacion+2),"b");
    circle(centrox*5+(radio/2),centroy-i*separacion,radio/2);
    outtextxy(centrox*5+(radio/4),centroy-12-i*separacion,"q6");

    outtextxy(centrox*6.5,centroy-i*(separacion+2),"r");
    circle(centrox*7+radio,centroy-i*separacion,radio/2);
    outtextxy(centrox*7+20,centroy-12-i*separacion,"q7");

    outtextxy(centrox*8.5,centroy-i*(separacion+2),"e");
    circle(centrox*9+radio,centroy-i*separacion,radio/2);
    outtextxy(centrox*9+20,centroy-12-i*separacion,"q8");

    outtextxy(centrox*10.5,centroy-i*(separacion+2),"a");
    circle(centrox*11+radio,centroy-i*separacion,radio/2);
    outtextxy(centrox*11+20,centroy-12-i*separacion,"q9");

    outtextxy(centrox*12.5,centroy-i*(separacion+2),"k");
    setcolor(BLUE);
    circle(centrox*13+radio,centroy-i*separacion,(radio/2)+2);
    circle(centrox*13+radio,centroy-i*separacion,(radio/2)+4);
    outtextxy(centrox*13+16,centroy-12-i*separacion,"q10");
}

```

```

setcolor(WHITE);
delay(time);

}
else if(i==4)
{

//c
outtextxy(centrox*4.5,centroy-i*(separacion+2),"c");
circle(centrox*5+(radio/2),centroy-i*separacion,radio/2);
outtextxy(centrox*5+(radio/4),centroy-12-i*separacion,"q11");

line(centrox*5+(radio/2),centroy-
i*separacion,centrox*11+(radio),centroy-(i*separacion));
//char
outtextxy(centrox*6.5,centroy-i*(separacion+2),"h");
circle(centrox*7+radio,centroy-i*separacion,radio/2);
outtextxy(centrox*7+20,centroy-12-i*separacion,"q15");

outtextxy(centrox*8.5,centroy-i*(separacion+2),"a");
circle(centrox*9+radio,centroy-i*separacion,radio/2);
outtextxy(centrox*9+20,centroy-12-i*separacion,"q16");

outtextxy(centrox*10.5,centroy-i*(separacion+2),"r");
setcolor(BLUE);
circle(centrox*11+radio,centroy-i*separacion,radio/2);
circle(centrox*11+radio,centroy-i*separacion,(radio/2)+2);
outtextxy(centrox*11+20,centroy-12-i*separacion,"q17");
setcolor(WHITE);
//case
line(centrox*5+(radio/2),centroy-
i*separacion,centrox*6+(radio),centroy-(i*separacion+20));
line(centrox*6+(radio),centroy-
(i*separacion+20),centrox*12+(radio),centroy-(i*separacion+20));

outtextxy(centrox*7,centroy-i*(separacion+8),"a");
circle(centrox*8.5,centroy-i*(separacion+5),radio/2);
outtextxy(centrox*8.5-10,centroy-12-i*(separacion+5),"q12");

outtextxy(centrox*9,centroy-i*(separacion+8),"s");
circle(centrox*10.5,centroy-i*(separacion+5),radio/2);
outtextxy(centrox*10.5-10,centroy-12-i*(separacion+5),"q13");

outtextxy(centrox*11,centroy-i*(separacion+8),"e");
setcolor(BLUE);
circle(centrox*12.5,centroy-i*(separacion+5),radio/2);
circle(centrox*12.5,centroy-i*(separacion+5),(radio/2)+2);
outtextxy(centrox*12.5-10,centroy-12-i*(separacion+5),"q14");
setcolor(WHITE);
//const
line(centrox*5+(radio/2),centroy-
i*separacion,centrox*6+(radio),centroy-(i*separacion-20));
line(centrox*6+(radio),centroy-(i*separacion-
20),centrox*25+(radio),centroy-(i*separacion-20));

outtextxy(centrox*7,centroy-i*(separacion-1),"o");
circle(centrox*19.5,centroy-i*(separacion-5),radio/2);
outtextxy(centrox*19.5-10,centroy-12-i*(separacion-5),"q18");

outtextxy(centrox*20.5,centroy-i*(separacion-1),"n");
circle(centrox*21.5,centroy-i*(separacion-5),radio/2);
outtextxy(centrox*21.5-10,centroy-12-i*(separacion-5),"q19");

outtextxy(centrox*22.5,centroy-i*(separacion-1),"s");
circle(centrox*23.5,centroy-i*(separacion-5),radio/2);
outtextxy(centrox*23.5-10,centroy-12-i*(separacion-5),"q25");

outtextxy(centrox*24.5,centroy-i*(separacion-1),"t");

```

```

setcolor(BLUE);
circle(centrox*25.5,centroy-i*(separacion-5),radio/2);
circle(centrox*25.5,centroy-i*(separacion-5),(radio/2)+2);
outtextxy(centrox*25.5-10,centroy-12-i*(separacion-5),"q26");
setcolor(WHITE);
//continue
line(centrox*21.5,centroy-i*(separacion-5),centrox*23.5,centroy-
i*(separacion+3));
line(centrox*23.5,centroy-i*(separacion+3),centrox*28,centroy-
i*(separacion+3));

outtextxy(centrox*23,centroy-i*(separacion+6),"t");
circle(centrox*23.5,centroy-i*(separacion+3),radio/2);
outtextxy(centrox*23.5-10,centroy-12-i*(separacion+3),"q20");

outtextxy(centrox*24,centroy-i*(separacion+6),"i");
circle(centrox*24.5,centroy-i*(separacion+3),radio/2);
outtextxy(centrox*24.5-10,centroy-12-i*(separacion+3),"q21");

outtextxy(centrox*25,centroy-i*(separacion+6),"n");
circle(centrox*25.5,centroy-i*(separacion+3),radio/2);
outtextxy(centrox*25.5-10,centroy-12-i*(separacion+3),"q22");

outtextxy(centrox*26,centroy-i*(separacion+6),"u");
circle(centrox*26.5,centroy-i*(separacion+3),radio/2);
outtextxy(centrox*26.5-10,centroy-12-i*(separacion+3),"q23");

outtextxy(centrox*27,centroy-i*(separacion+6),"e");
setcolor(BLUE);
circle(centrox*28,centroy-i*(separacion+3),radio/2);
circle(centrox*28,centroy-i*(separacion+3),(radio/2)+2);
outtextxy(centrox*28-10,centroy-12-i*(separacion+3),"q24");
setcolor(WHITE);
delay(time);
}
else if(i==3)
{

//do, double
line(centrox*5+(radio/2),centroy-
i*separacion,centrox*15+(radio),centroy-i*(separacion));
outtextxy(centrox*4.5,centroy-i*(separacion+2),"d");
circle(centrox*5+(radio/2),centroy-i*separacion,radio/2);
outtextxy(centrox*5+(radio/4),centroy-12-i*separacion,"q27");

outtextxy(centrox*6.5,centroy-i*(separacion+2),"o");
setcolor(BLUE);
circle(centrox*7+radio,centroy-i*separacion,(radio/2)-1);
circle(centrox*7+radio,centroy-i*separacion,(radio/2));
outtextxy(centrox*7+20,centroy-12-i*separacion,"q34");
setcolor(WHITE);

outtextxy(centrox*8.5,centroy-i*(separacion+2),"u");
circle(centrox*9+radio,centroy-i*separacion,(radio/2)-1);
outtextxy(centrox*9+20,centroy-12-i*separacion,"q35");

outtextxy(centrox*10.5,centroy-i*(separacion+2),"b");
circle(centrox*11+radio,centroy-i*separacion,(radio/2)-1);
outtextxy(centrox*11+20,centroy-12-i*separacion,"q36");

outtextxy(centrox*12.5,centroy-i*(separacion+2),"l");
circle(centrox*13+radio,centroy-i*separacion,(radio/2)-1);
outtextxy(centrox*13+20,centroy-12-i*separacion,"q37");

outtextxy(centrox*14.5,centroy-i*(separacion+2),"e");
setcolor(BLUE);
circle(centrox*15+radio,centroy-i*separacion,(radio/2));

```

```

circle(centrox*15+radio,centroy-i*separacion,(radio/2)-1);
outtextxy(centrox*15+20,centroy-12-i*separacion,"q38");
setcolor(WHITE);
//default
line(centrox*5+(radio/2),centroy-
i*separacion,centrox*6+(radio),centroy-(i*separacion+20));
line(centrox*6+(radio),centroy-
(i*separacion+20),centrox*18+(radio),centroy-(i*separacion+20));

outtextxy(centrox*7,centroy-i*(separacion+10),"e");
circle(centrox*8.5,centroy-i*(separacion+5),(radio/2)-1);
outtextxy(centrox*8.5-20,centroy-12-i*(separacion+5),"q28");

outtextxy(centrox*9,centroy-i*(separacion+10),"f");
circle(centrox*10.5,centroy-i*(separacion+5),(radio/2)-1);
outtextxy(centrox*10.5-20,centroy-12-i*(separacion+5),"q29");

outtextxy(centrox*11,centroy-i*(separacion+10),"a");
circle(centrox*12.5,centroy-i*(separacion+5),(radio/2)-1);
outtextxy(centrox*12.5-20,centroy-12-i*(separacion+5),"q30");

outtextxy(centrox*13,centroy-i*(separacion+10),"u");
circle(centrox*14.5,centroy-i*(separacion+5),(radio/2)-1);
outtextxy(centrox*14.5-20,centroy-12-i*(separacion+5),"q31");

outtextxy(centrox*15,centroy-i*(separacion+10),"l");
circle(centrox*16.5,centroy-i*(separacion+5),(radio/2)-1);
outtextxy(centrox*16.5-20,centroy-12-i*(separacion+5),"q32");

outtextxy(centrox*17,centroy-i*(separacion+10),"t");
setcolor(BLUE);
circle(centrox*18.5,centroy-i*(separacion+5),(radio/2)-1);
circle(centrox*18.5,centroy-i*(separacion+5),(radio/2));
outtextxy(centrox*18.5-20,centroy-12-i*(separacion+5),"q33");
setcolor(WHITE);

delay(time);

}
else if(i==2)
{

//extern
line(centrox*5+(radio/2),centroy-
i*separacion,centrox*15+(radio),centroy-(i*separacion));
outtextxy(centrox*4.5,centroy-i*(separacion+6),"e");
circle(centrox*5+(radio/2),centroy-i*separacion,radio/2);
outtextxy(centrox*5+(radio/4),centroy-12-i*separacion,"q39");

outtextxy(centrox*6.5,centroy-i*(separacion+6),"x");
circle(centrox*7+radio,centroy-i*separacion,radio/2);
outtextxy(centrox*7+20,centroy-12-i*separacion,"q43");

outtextxy(centrox*8.5,centroy-i*(separacion+6),"t");
circle(centrox*9+radio,centroy-i*separacion,radio/2);
outtextxy(centrox*9+20,centroy-12-i*separacion,"q44");

outtextxy(centrox*10.5,centroy-i*(separacion+6),"e");
circle(centrox*11+radio,centroy-i*separacion,radio/2);
outtextxy(centrox*11+20,centroy-12-i*separacion,"q45");

outtextxy(centrox*12.5,centroy-i*(separacion+6),"r");
circle(centrox*13+radio,centroy-i*separacion,radio/2);
outtextxy(centrox*13+20,centroy-12-i*separacion,"q46");

outtextxy(centrox*14.5,centroy-i*(separacion+6),"n");
setcolor(BLUE);

```

```

circle(centrox*15+radio,centroy-i*separacion,(radio/2)+2);
circle(centrox*15+radio,centroy-i*separacion,(radio/2)+4);
outtextxy(centrox*15+16,centroy-12-i*separacion,"q47");
setcolor(WHITE);
//else
line(centrox*5+(radio/2),centroy-
i*separacion,centrox*6+(radio),centroy-(i*separacion+20));
line(centrox*6+(radio),centroy-
(i*separacion+20),centrox*20.5+(radio),centroy-(i*separacion+20));

outtextxy(centrox*16,centroy-i*(separacion+15),"l");
circle(centrox*17,centroy-i*(separacion+10),radio/2);
outtextxy(centrox*17-10,centroy-12-i*(separacion+10),"q40");

outtextxy(centrox*18,centroy-i*(separacion+15),"s");
circle(centrox*19,centroy-i*(separacion+10),radio/2);
outtextxy(centrox*19-10,centroy-12-i*(separacion+10),"q41");

outtextxy(centrox*20,centroy-i*(separacion+15),"e");
setcolor(BLUE);
circle(centrox*21,centroy-i*(separacion+10),radio/2);
circle(centrox*21,centroy-i*(separacion+10),(radio/2)+2);
outtextxy(centrox*21-10,centroy-12-i*(separacion+10),"q42");
setcolor(WHITE);

//enum
line(centrox*5+(radio/2),centroy-
i*separacion,centrox*6+(radio),centroy-(i*separacion-20));
line(centrox*6+(radio),centroy-(i*separacion-
20),centrox*20.5+(radio),centroy-(i*separacion-20));

outtextxy(centrox*16,centroy-i*(separacion-4),"n");
circle(centrox*17,centroy-i*(separacion-10),radio/2);
outtextxy(centrox*17-10,centroy-12-i*(separacion-10),"q48");

outtextxy(centrox*18,centroy-i*(separacion-4),"u");
circle(centrox*19,centroy-i*(separacion-10),radio/2);
outtextxy(centrox*19-10,centroy-12-i*(separacion-10),"q49");

outtextxy(centrox*20,centroy-i*(separacion-4),"m");
setcolor(BLUE);
circle(centrox*21,centroy-i*(separacion-10),radio/2);
circle(centrox*21,centroy-i*(separacion-10),(radio/2)+2);
outtextxy(centrox*21-10,centroy-12-i*(separacion-10),"q50");
setcolor(WHITE);

delay(time);
}
else if(i==1)
{

//float
line(centrox*5+(radio/2),centroy-
i*separacion,centrox*13+(radio),centroy-(i*separacion));
outtextxy(centrox*4.5,centroy-i*(separacion+10),"f");
circle(centrox*5+(radio/2),centroy-i*separacion,radio/2);
outtextxy(centrox*5+(radio/4),centroy-12-i*separacion,"q51");

outtextxy(centrox*6.5,centroy-i*(separacion+10),"l");
circle(centrox*7+radio,centroy-i*separacion,radio/2);
outtextxy(centrox*7+20,centroy-12-i*separacion,"q52");

outtextxy(centrox*8.5,centroy-i*(separacion+10),"o");
circle(centrox*9+radio,centroy-i*separacion,radio/2);
outtextxy(centrox*9+20,centroy-12-i*separacion,"q53");

outtextxy(centrox*10.5,centroy-i*(separacion+10),"a");

```



```

        circle(centrox*11+radio,centroy-i*separacion,radio/2);
        outtextxy(centrox*11+20,centroy-12-i*separacion,"q54");

        outtextxy(centrox*12.5,centroy-i*(separacion+10),"t");
        setcolor(BLUE);
        circle(centrox*13+radio,centroy-i*separacion,(radio/2)+2);
        circle(centrox*13+radio,centroy-i*separacion,(radio/2)+4);
        outtextxy(centrox*13+16,centroy-12-i*separacion,"q55");
        setcolor(WHITE);
        //for
        line(centrox*5+(radio/2),centroy-
i*separacion,centrox*6+(radio),centroy-(i*separacion-20));
        line(centrox*6+(radio),centroy-(i*separacion-
20),centrox*16.5+(radio),centroy-(i*separacion-20));

        outtextxy(centrox*7,centroy-i*(separacion-5),"o");
        circle(centrox*15,centroy-i*(separacion-20),radio/2);
        outtextxy(centrox*15-10,centroy-12-i*(separacion-20),"q56");

        outtextxy(centrox*16,centroy-i*(separacion-5),"r");
        setcolor(BLUE);
        circle(centrox*17,centroy-i*(separacion-20),(radio/2)+2);
        circle(centrox*17,centroy-i*(separacion-20),radio/2);
        outtextxy(centrox*17-10,centroy-12-i*(separacion-20),"q57");
        setcolor(WHITE);

        delay(time);
    }

}

//int,if
line(centrox+radio,centroy,centrox*3,centroy);
line(centrox*3,centroy,centrox*9,centroy);

outtextxy(centrox*4.5,centroy-10,"i");
circle(centrox*5+(radio/2),centroy,radio/2);
outtextxy(centrox*5,centroy-10,"q58");

outtextxy(centrox*6.5,centroy-10,"n");
circle(centrox*7+(radio/2),centroy,radio/2);
outtextxy(centrox*7,centroy-10,"q59");

outtextxy(centrox*8.5,centroy-10,"t");
setcolor(BLUE);
circle(centrox*9+(radio/2),centroy,radio/2);
circle(centrox*9+(radio/2),centroy,(radio/2)+2);
outtextxy(centrox*9,centroy-10,"q60");
setcolor(WHITE);

line(centrox*5+(radio/2),centroy,centrox*6+(radio),centroy+20);
line(centrox*6+(radio),centroy+20,centrox*10+(radio),centroy+20);

outtextxy(centrox*6.7,centroy+8,"f");
setcolor(BLUE);
circle(centrox*10.5,centroy+15,radio/2);
circle(centrox*10.5,centroy+15,(radio/2)+2);
outtextxy(centrox*10.2,centroy+5,"q61");
setcolor(WHITE);

delay(time);
for(int i=1; i<=7; i++){
    line(centrox+radio,centroy,centrox*3,centroy+i*separacion);
    line(centrox*3,centroy+i*separacion,centrox*5,centroy+i*separacion);
    delay(time);
    if(i==7)
    {
        //while

```

```

line(centrox*5+radio,centroy+i*separacion,centrox*13+radio,centroy+i*separacion);
    outtextxy(centrox*4.5,centroy+i*(separacion-3),"w");
    circle(centrox*5+(radio/2),centroy+i*separacion,radio/2);
    outtextxy(centrox*5+(radio/4),centroy-10+i*separacion,"q130");

line(centrox*13+radio,centroy+i*separacion,centrox*13+radio,centroy+6*separacion);
    outtextxy(centrox*6.5+(radio/4),centroy-10+i*separacion,"h");
}
else if(i==6)
{

    //volatile

line(centrox*5+radio,centroy+i*separacion,centrox*19+radio,centroy+i*separacion);
    outtextxy(centrox*4.5,centroy+i*(separacion-2),"v");
    circle(centrox*5+(radio/2),centroy+i*separacion,radio/2);
    outtextxy(centrox*5+(radio/4),centroy-10+i*separacion,"q120");

    outtextxy(centrox*6.5,centroy+i*(separacion-2),"o");
    circle(centrox*7+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*7+20,centroy-10+i*separacion,"q121");

    outtextxy(centrox*8.5,centroy+i*(separacion-2),"l");
    circle(centrox*9+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*9+20,centroy-10+i*separacion,"q124");

    outtextxy(centrox*10.5,centroy+i*(separacion-2),"a");
    circle(centrox*11+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*11+20,centroy-10+i*separacion,"q125");

    outtextxy(centrox*12.5,centroy+i*(separacion-2),"t");
    circle(centrox*13+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*13+20,centroy-10+i*separacion,"q126");

    outtextxy(centrox*14.5,centroy+i*(separacion-2),"i");
    circle(centrox*15+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*15+20,centroy-10+i*separacion,"q127");

    outtextxy(centrox*16.5,centroy+i*(separacion-2),"l");
    circle(centrox*17+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*17+20,centroy-10+i*separacion,"q128");

    outtextxy(centrox*18.5,centroy+i*(separacion-2),"e");
    setcolor(BLUE);
    circle(centrox*19+radio,centroy+i*separacion,(radio/2)+2);
    circle(centrox*19+radio,centroy+i*separacion,(radio/2)+4);
    outtextxy(centrox*19+20,centroy-10+i*separacion,"q129");
    setcolor(WHITE);

    //void

line(centrox*7+radio,centroy+i*separacion,centrox*8+(radio),centroy+(i*separacion-20));
    line(centrox*8+(radio),centroy+(i*separacion-20),centrox*22.5+(radio),centroy+(i*separacion-20));

    outtextxy(centrox*9,centroy+i*(separacion-5),"i");
    circle(centrox*21,centroy+i*(separacion-3),radio/2);
    outtextxy(centrox*21-10,centroy-12+i*(separacion-3),"q122");

    outtextxy(centrox*22,centroy+i*(separacion-5),"d");
    setcolor(BLUE);
    circle(centrox*23,centroy+i*(separacion-3),radio/2);
    circle(centrox*23,centroy+i*(separacion-3),(radio/2)+2);
    outtextxy(centrox*23-10,centroy-12+i*(separacion-3),"q123");
    setcolor(WHITE);
}

```

```

else if(i==5)
{
    //typedef

    line(centrox*5+radio,centroy+i*separacion,centrox*17+radio,centroy+i*separacion);
    outtextxy(centrox*4.5,centroy+i*(separacion-2),"t");
    circle(centrox*5+(radio/2),centroy+i*separacion,radio/2);
    outtextxy(centrox*5+(radio/4),centroy-10+i*separacion,"q113");

    outtextxy(centrox*6.5,centroy+i*(separacion-2),"y");
    circle(centrox*7+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*7+20,centroy-10+i*separacion,"q114");

    outtextxy(centrox*8.5,centroy+i*(separacion-2),"p");
    circle(centrox*9+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*9+20,centroy-10+i*separacion,"q115");

    outtextxy(centrox*10.5,centroy+i*(separacion-2),"e");
    circle(centrox*11+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*11+20,centroy-10+i*separacion,"q116");

    outtextxy(centrox*12.5,centroy+i*(separacion-2),"d");
    circle(centrox*13+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*13+20,centroy-10+i*separacion,"q117");

    outtextxy(centrox*14.5,centroy+i*(separacion-2),"e");
    circle(centrox*15+radio,centroy+i*separacion,radio/2);
    outtextxy(centrox*15+20,centroy-10+i*separacion,"q118");

    outtextxy(centrox*16.5,centroy+i*(separacion-2),"f");
    setcolor(BLUE);
    circle(centrox*17+radio,centroy+i*separacion,(radio/2)+2);
    circle(centrox*17+radio,centroy+i*separacion,(radio/2)+4);
    outtextxy(centrox*17+20,centroy-10+i*separacion,"q119");
    setcolor(WHITE);
    delay(time);
}
else if(i==4)
{
    //static

    line(centrox*5+radio,centroy+i*separacion,centrox*28+radio,centroy+i*separacion);
    outtextxy(centrox*4.5,centroy+i*(separacion-3),"s");
    circle(centrox*5+(radio/2),centroy+i*separacion,radio/2);
    outtextxy(centrox*5+(radio/4),centroy-10+i*separacion,"q85");

    outtextxy(centrox*6.5,centroy+i*(separacion-3),"t");
    circle(centrox*22+(radio/2),centroy+i*separacion,radio/2);
    outtextxy(centrox*22+(radio/4),centroy-10+i*separacion,"q95");

    outtextxy(centrox*23,centroy+i*(separacion-3),"a");
    circle(centrox*23.5+(radio/2),centroy+i*separacion,radio/2);
    outtextxy(centrox*23.5+(radio/4),centroy-10+i*separacion,"q96");

    outtextxy(centrox*24.5,centroy+i*(separacion-3),"t");
    circle(centrox*25+(radio/2),centroy+i*separacion,radio/2);
    outtextxy(centrox*25+(radio/4),centroy-10+i*separacion,"q97");

    outtextxy(centrox*26,centroy+i*(separacion-3),"i");
    circle(centrox*26.5+(radio/2),centroy+i*separacion,radio/2);
    outtextxy(centrox*26.5+(radio/4),centroy-10+i*separacion,"q98");

    outtextxy(centrox*27.5,centroy+i*(separacion-3),"c");
    setcolor(BLUE);
    circle(centrox*28+(radio/2),centroy+i*separacion,radio/2);
    circle(centrox*28+(radio/2),centroy+i*separacion,(radio/2)+2);

```

```

        outtextxy(centrox*28+(radio/4),centroy-10+i*separacion,"q99");
        setcolor(WHITE);
        //struct

        line(centrox*22+(radio/2),centroy+i*separacion,centrox*23+(radio/2),centroy+i*separacion+30);

        line(centrox*23+(radio/2),centroy+i*separacion+30,centrox*28.5+(radio/2),centroy+i*separacion+30);

        outtextxy(centrox*23.5+(radio/2),centroy+i*separacion+20,"r");
        circle(centrox*24.5+(radio/2),centroy+i*separacion+30,radio/2);
        outtextxy(centrox*24.5+(radio/4),centroy-
10+i*separacion+30,"q100");

        outtextxy(centrox*25.2+(radio/2),centroy+i*separacion+20,"u");
        circle(centrox*26+(radio/2),centroy+i*separacion+30,radio/2);
        outtextxy(centrox*26+(radio/4),centroy-
10+i*separacion+30,"q101");

        outtextxy(centrox*26.7+(radio/2),centroy+i*separacion+20,"c");
        circle(centrox*27.4+(radio/2),centroy+i*separacion+30,radio/2);
        outtextxy(centrox*27.4+(radio/4),centroy-
10+i*separacion+30,"q102");

        outtextxy(centrox*28.1+(radio/2),centroy+i*separacion+20,"t");
        setcolor(BLUE);
        circle(centrox*28.9+(radio/2),centroy+i*separacion+30,radio/2);

        circle(centrox*28.9+(radio/2),centroy+i*separacion+30,(radio/2)+2);
        outtextxy(centrox*28.9+(radio/4),centroy-
10+i*separacion+30,"q103");

        setcolor(WHITE);
        //sizeof

        line(centrox*5+radio,centroy+i*separacion,centrox*6+(radio),centroy+(i*separacion-20));
        line(centrox*6+(radio),centroy+(i*separacion-
20),centrox*25+(radio),centroy+(i*separacion-20));

        outtextxy(centrox*7,centroy+i*(separacion-8),"i");
        circle(centrox*9,centroy+i*(separacion-5),radio/2);
        outtextxy(centrox*9-10,centroy-12+i*(separacion-5),"q86");

        outtextxy(centrox*15,centroy+i*(separacion-8),"z");
        circle(centrox*17,centroy+i*(separacion-5),radio/2);
        outtextxy(centrox*17-10,centroy-12+i*(separacion-5),"q91");

        outtextxy(centrox*18,centroy+i*(separacion-8),"e");
        circle(centrox*20,centroy+i*(separacion-5),radio/2);
        outtextxy(centrox*20-10,centroy-12+i*(separacion-5),"q92");

        outtextxy(centrox*21,centroy+i*(separacion-8),"o");
        circle(centrox*23,centroy+i*(separacion-5),radio/2);
        outtextxy(centrox*23-10,centroy-12+i*(separacion-5),"q93");

        outtextxy(centrox*24,centroy+i*(separacion-8),"f");
        setcolor(BLUE);
        circle(centrox*26,centroy+i*(separacion-5),radio/2);
        circle(centrox*26,centroy+i*(separacion-5),(radio/2)+2);
        outtextxy(centrox*26-10,centroy-12+i*(separacion-5),"q94");

        setcolor(WHITE);
        //signed
        line(centrox*13+(radio),centroy+(i*separacion-
20),centrox*14+(radio),centroy+(i*separacion-40));
        line(centrox*14+(radio),centroy+(i*separacion-
40),centrox*24.5+(radio),centroy+(i*separacion-40));

        outtextxy(centrox*14+(radio),centroy+(i*separacion-49),"g");

```

```

circle(centrox*16,centroy+i*(separacion-10),radio/2);
outtextxy(centrox*16-10,centroy-12+i*(separacion-10),"q87");

outtextxy(centrox*17+(radio),centroy+(i*separacion-49),"n");
circle(centrox*19,centroy+i*(separacion-10),radio/2);
outtextxy(centrox*19-10,centroy-12+i*(separacion-10),"q88");

outtextxy(centrox*20+(radio),centroy+(i*separacion-49),"e");
circle(centrox*22,centroy+i*(separacion-10),radio/2);
outtextxy(centrox*22-10,centroy-12+i*(separacion-10),"q89");

outtextxy(centrox*23+(radio),centroy+(i*separacion-49),"d");
setcolor(BLUE);
circle(centrox*25,centroy+i*(separacion-10),radio/2);
circle(centrox*25,centroy+i*(separacion-10),(radio/2)+2);
outtextxy(centrox*25-10,centroy-12+i*(separacion-10),"q89");
setcolor(WHITE);
//switch

line(centrox*5+radio,centroy+i*separacion,centrox*6+(radio),centroy+(i*separacion+15));

line(centrox*6+(radio),centroy+(i*separacion+15),centrox*18+(radio),centroy+(i*separacion+15));

outtextxy(centrox*7+(radio),centroy+(i*separacion+5),"w");
circle(centrox*9,centroy+i*(separacion+4),radio/2);
outtextxy(centrox*9-10,centroy-12+i*(separacion+4),"q108");

outtextxy(centrox*9.3+(radio),centroy+(i*separacion+5),"i");
circle(centrox*11.3,centroy+i*(separacion+4),radio/2);
outtextxy(centrox*11.3-10,centroy-12+i*(separacion+4),"q109");

outtextxy(centrox*11.6+(radio),centroy+(i*separacion+5),"t");
circle(centrox*13.6,centroy+i*(separacion+4),radio/2);
outtextxy(centrox*13.6-10,centroy-12+i*(separacion+4),"q110");

outtextxy(centrox*14+(radio),centroy+(i*separacion+5),"c");
circle(centrox*16,centroy+i*(separacion+4),radio/2);
outtextxy(centrox*16-10,centroy-12+i*(separacion+4),"q111");

outtextxy(centrox*16.3+(radio),centroy+(i*separacion+5),"h");
setcolor(BLUE);
circle(centrox*18.3,centroy+i*(separacion+4),radio/2);
circle(centrox*18.3,centroy+i*(separacion+4),(radio/2)+2);
outtextxy(centrox*18.3-10,centroy-12+i*(separacion+4),"q112");
setcolor(WHITE);
//short

line(centrox*5+radio,centroy+i*separacion,centrox*6+(radio),centroy+(i*separacion+30));

line(centrox*6+(radio),centroy+(i*separacion+30),centrox*21+(radio),centroy+(i*separacion+30));

outtextxy(centrox*7.5+(radio),centroy+(i*separacion+18),"h");
circle(centrox*10.5,centroy+i*(separacion+8),radio/2);
outtextxy(centrox*10.5-10,centroy-12+i*(separacion+8),"q104");

outtextxy(centrox*12+(radio),centroy+(i*separacion+18),"o");
circle(centrox*19,centroy+i*(separacion+8),radio/2);
outtextxy(centrox*19-10,centroy-12+i*(separacion+8),"q105");

outtextxy(centrox*19.2+(radio),centroy+(i*separacion+18),"r");
circle(centrox*20.5,centroy+i*(separacion+8),radio/2);
outtextxy(centrox*20.5-10,centroy-12+i*(separacion+8),"q106");

outtextxy(centrox*20.6+(radio),centroy+(i*separacion+18),"t");
setcolor(BLUE);
circle(centrox*22,centroy+i*(separacion+8),radio/2);
circle(centrox*22,centroy+i*(separacion+8),(radio/2)+2);

```

```

        outtextxy(centrox*22-10,centroy-12+i*(separacion+8),"q107");
        setcolor(WHITE);
    }
    else if(i==3)
    {

        //unsigned

        line(centrox*5+radio,centroy+i*separacion,centrox*15.5+radio,centroy+i*separacion);
        outtextxy(centrox*4.5,centroy+i*(separacion-3),"u");
        circle(centrox*5+(radio/2),centroy+i*separacion,radio/2);
        outtextxy(centrox*5+(radio/4),centroy-10+i*separacion,"q78");

        outtextxy(centrox*6.5,centroy+i*(separacion-3),"n");
        circle(centrox*7+radio,centroy+i*separacion,radio/2);
        outtextxy(centrox*7+20,centroy-10+i*separacion,"q79");

        outtextxy(centrox*8.5,centroy+i*(separacion-2),"s");
        circle(centrox*9+radio,centroy+i*separacion,radio/2);
        outtextxy(centrox*9+20,centroy-10+i*separacion,"q83");

        outtextxy(centrox*10.5,centroy+i*(separacion-2),"i");
        circle(centrox*11+radio,centroy+i*separacion,radio/2);
        outtextxy(centrox*11+20,centroy-10+i*separacion,"q84");

        outtextxy(centrox*12.5,centroy+i*(separacion-2),"g");
        //union

        line(centrox*7+radio,centroy+i*separacion,centrox*8+(radio),centroy+(i*separacion-20));
        line(centrox*8+(radio),centroy+(i*separacion-
20),centrox*20+(radio),centroy+(i*separacion-20));

        outtextxy(centrox*9,centroy+i*(separacion-10),"i");
        circle(centrox*17,centroy+i*(separacion-7),radio/2);
        outtextxy(centrox*17-10,centroy-12+i*(separacion-7),"q80");

        outtextxy(centrox*18,centroy+i*(separacion-10),"o");
        circle(centrox*19,centroy+i*(separacion-7),radio/2);
        outtextxy(centrox*19-10,centroy-12+i*(separacion-7),"q81");

        outtextxy(centrox*20,centroy+i*(separacion-10),"n");
        setcolor(BLUE);
        circle(centrox*21,centroy+i*(separacion-7),radio/2);
        circle(centrox*21,centroy+i*(separacion-7),(radio/2)+2);
        outtextxy(centrox*21-10,centroy-12+i*(separacion-7),"q82");
        setcolor(WHITE);

    }
    else if(i==2)
    {

        //return

        line(centrox*5+radio,centroy+i*separacion,centrox*15+radio,centroy+i*separacion);
        outtextxy(centrox*4.5,centroy+i*(separacion-5),"r");
        circle(centrox*5+(radio/2),centroy+i*separacion,radio/2);
        outtextxy(centrox*5+(radio/4),centroy-10+i*separacion,"q66");

        outtextxy(centrox*6.5,centroy+i*(separacion-5),"e");
        circle(centrox*7+radio,centroy+i*separacion,radio/2);
        outtextxy(centrox*7+20,centroy-10+i*separacion,"q67");

        outtextxy(centrox*8.5,centroy+i*(separacion-5),"t");
        circle(centrox*9+radio,centroy+i*separacion,radio/2);
        outtextxy(centrox*9+20,centroy-10+i*separacion,"q74");

        outtextxy(centrox*10.5,centroy+i*(separacion-5),"u");

```

```

circle(centrox*11+radio,centroy+i*separacion,radio/2);
outtextxy(centrox*11+20,centroy-10+i*separacion,"q75");

outtextxy(centrox*12.5,centroy+i*(separacion-5),"r");
circle(centrox*13+radio,centroy+i*separacion,radio/2);
outtextxy(centrox*13+20,centroy-10+i*separacion,"q76");

outtextxy(centrox*14.5,centroy+i*(separacion-5),"n");
setcolor(BLUE);
circle(centrox*15+radio,centroy+i*separacion,(radio/2)+2);
circle(centrox*15+radio,centroy+i*separacion,(radio/2)+4);
outtextxy(centrox*15+20,centroy-10+i*separacion,"q77");
setcolor(WHITE);
//register

line(centrox*7+radio,centroy+i*separacion,centrox*8+(radio),centroy+(i*separacion-20));
line(centrox*8+(radio),centroy+(i*separacion-20),centrox*26.5+(radio),centroy+(i*separacion-20));

outtextxy(centrox*9,centroy+i*(separacion-18),"g");
circle(centrox*17,centroy+i*(separacion-10),radio/2);
outtextxy(centrox*17-10,centroy-12+i*(separacion-10),"q68");

outtextxy(centrox*18,centroy+i*(separacion-18),"i");
circle(centrox*19,centroy+i*(separacion-10),radio/2);
outtextxy(centrox*19-10,centroy-12+i*(separacion-10),"q69");

outtextxy(centrox*20,centroy+i*(separacion-18),"s");
circle(centrox*21,centroy+i*(separacion-10),radio/2);
outtextxy(centrox*21-10,centroy-12+i*(separacion-10),"q70");

outtextxy(centrox*22,centroy+i*(separacion-18),"t");
circle(centrox*23,centroy+i*(separacion-10),radio/2);
outtextxy(centrox*23-10,centroy-12+i*(separacion-10),"q71");

outtextxy(centrox*24,centroy+i*(separacion-18),"e");
circle(centrox*25,centroy+i*(separacion-10),radio/2);
outtextxy(centrox*25-10,centroy-12+i*(separacion-10),"q72");

outtextxy(centrox*26,centroy+i*(separacion-18),"r");
setcolor(BLUE);
circle(centrox*27,centroy+i*(separacion-10),radio/2);
circle(centrox*27,centroy+i*(separacion-10),(radio/2)+2);
outtextxy(centrox*27-10,centroy-12+i*(separacion-10),"q73");
setcolor(WHITE);

}
else if(i==1)
{

//long

line(centrox*5+radio,centroy+i*separacion,centrox*11+radio,centroy+i*separacion);
outtextxy(centrox*4.5,centroy+i*(separacion-10),"l");
circle(centrox*5+(radio/2),centroy+i*separacion,radio/2);
outtextxy(centrox*5+(radio/4),centroy-10+i*separacion,"q62");

outtextxy(centrox*6.5,centroy+i*(separacion-10),"o");
circle(centrox*7+radio,centroy+i*separacion,radio/2);
outtextxy(centrox*7+20,centroy-10+i*separacion,"q63");

outtextxy(centrox*8.5,centroy+i*(separacion-10),"n");
circle(centrox*9+radio,centroy+i*separacion,radio/2);
outtextxy(centrox*9+20,centroy-10+i*separacion,"q64");

```

```
        outtextxy(centrox*10.5,centroy+i*(separacion-10),"g");
        setcolor(BLUE);
        circle(centrox*11+radio,centroy+i*separacion,(radio/2)+2);
        circle(centrox*11+radio,centroy+i*separacion,(radio/2)+4);
        outtextxy(centrox*11+20,centroy-10+i*separacion,"q65");
        setcolor(WHITE);
        delay(time);
    }

    }
    abrirArchivo();
    delay(10000);
}
```