```
/*
 Compiladores - Analizador Lexico para el lenguaje C#

 Facultad de Ciencias de la Computacion - B.U.A.P - Verano 2004
 Profesor: Pedro Bello Lopez.
 Alumno: Javier B. Camacho Martinez. Matricula: 990002830
 E-mail: <vaLar@unixmexico.org>

 Este programa lee un archivo fuente (escrito en C#) y calcula las parejas
 ordenadas <token, lexema> por medio de un automata finito determinista.
 Los tokens son guardados en un archivo de texto. (tokens.txt)

 Este programa esta diseñado para compilarse en Turbo C de Borland.

 Licencia:
 Este programa es software libre, por lo tanto se puede distribuir o
 modificar bajo los terminos de la licencia GNU (General Public License)
 asi como lo ha publicado la fundacion de software libre.

 Si se realiza algun tipo de mejora o cambio significativo a este programa,
 favor de notificar por e-mail a el autor

 Si tiene algun reporte de bugs, comentarios, preguntas o sugerencias
 envielas por mail a: valar@unixmexico.org
*/

#include<math.h>
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<ctype.h>
#include<string.h>

int MT[94][30];
void llenar_matriz();
void corrige_arreglo(char arre[100]);

void main() {

FILE *apt, *apt2;
char arch[15];
char c;
char arre[100];
int ind=0;
int ban=0, ban2=0;
int edo=1;
int ptr;
int yesp;

char *palabras[] = {"abstract", "as", "bool", "break", "byte", "case",
"catch", "char", "checked", "class", "const", "continue", "decimal",
"default", "delegate", "do", "double", "else", "enum", "event", "explicit",
"extern", "false", "finally", "fixed", "float", "for", "foreach", "goto",
"if", "implicit", "in", "int", "interface", "internal", "lock", "is", "long",
"namespace", "new", "null", "object", "operator", "out", "override", "params",
"private", "protect", "public", "readonly", "ref", "return", "sbyte", "sealed",
"short", "sizeof", "stackalloc", "static", "string", "struct", "switch","this",
"throw", "true", "try","typeof", "uint", "ulong", "unchecked", "unsafe",
"ushort", "using", "virtual", "void", "while"};

strncpy(arre,"",100);

printf("Analizador lexico para C#\n");
printf("Escriba el nombre del archivo que quiere analizar: ");
scanf("%s",arch);
printf("Intentando abrir: %s\n\n",arch);
apt=fopen(arch,"r");
apt2=fopen("tokens.txt","w");
```

```
if(apt == 0){
   printf("No se encontro el archivo\n");
   getch();
   exit(0);
   }

llenar_matriz();

while(!feof(apt)){
 if(ban==0) c=getc(apt);
 ban=0;
    // Es numero o letra?
 if(isdigit(c))  edo=MT[edo][28];
 if(isalpha(c))  edo=MT[edo][6];

   // Comparacion del caracter de entrada!!
 switch(c){
  case '+':edo=MT[edo][1];break;
  case '-':edo=MT[edo][2];break;
  case '*':edo=MT[edo][3];break;
  case '/':edo=MT[edo][4];break;
  case '.':edo=MT[edo][5];break;
  case '=':edo=MT[edo][7];break;
  case '!':edo=MT[edo][8];break;
  case '%':edo=MT[edo][9];break;
  case '<':edo=MT[edo][10];break;
  case '>':edo=MT[edo][11];break;
  case '^':edo=MT[edo][12];break;
  case '|':edo=MT[edo][13];break;
  case '&':edo=MT[edo][14];break;
  case '[':edo=MT[edo][15];break;
  case ']':edo=MT[edo][16];break;
  case '(':edo=MT[edo][17];break;
  case ')':edo=MT[edo][18];break;
  case '{':edo=MT[edo][19];break;
  case '}':edo=MT[edo][20];break;
  case '\"':edo=MT[edo][21];break;
  case ';':edo=MT[edo][22];break;
  case ',':edo=MT[edo][23];break;
  case '#':edo=MT[edo][24];break;
  case ' ':edo=MT[edo][25];break;
  case '_':edo=MT[edo][26];break;
  case '\'':edo=MT[edo][27];break;
  case '\n':edo=MT[edo][29];break;
  case ':' :edo=MT[edo][30];break;
 } //fin switch

  // Estados terminales del automata!!
 switch (edo) {
  case 3:    // SumaAsigna
       edo=1;
       corrige_arreglo(arre);
       fprintf(apt2,"SumaAsigna,    %s\n",arre);
       strncpy(arre,"",100);
       ind=0;
       ban=1;
       break;

  case 4:    // *Suma
       edo=1;
       corrige_arreglo(arre);
       fprintf(apt2,"Suma,          %s\n",arre);
       strncpy(arre,"",100);
       ind=0;
       ban=1;
       break;
```

```
case 5:    // Incremento
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Incremento,    %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 7:    // RestaAsigna
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"RestaAsigna,   %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 8:    // *Resta
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Resta,         %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 9:    // Decremento
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Decremento,    %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 11:    // MultiplicaAsigna
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"MultiplicaAsigna, %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 12:    // *Multiplica
     edo=1;
     fprintf(apt2,"Multiplica,    %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 13:    // FinComentario
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"FinComentario, %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 15:    // *Divide
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Divide,        %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
```

```
     ban=1;
     break;

case 16:    // *DivideAsigna
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"DivideAsigna,  %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 18:    // ComentarioLinea
     edo=1;
     corrige_arreglo(arre);
     // fprintf(apt2,"ComentarioLinea,  %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 22:    // *Comentario
     edo=1;
     // fprintf(apt2,"Comentario, %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 24:    // Identico
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Identico,           %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 25:    // *Asigna
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Asigna,        %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 27:    // Diferente
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Diferente,     %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 28:    // *Negacion
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Negacion,           %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 30:    // ModuloAsigna
     edo=1;
     corrige_arreglo(arre);
```

```
     fprintf(apt2,"ModuloAsigna,  %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 31:    // *Modulo
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Modulo,        %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 33:    // MenorIgual
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"MenorIgual,    %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 34:    // *MenorQue
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"MenorQue,           %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 36:    // *CorrimientoIzq
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"CorrimientoIzq,     %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 37:    // *CorrimientoIzqAsigna
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"CorrimientoIzqAsigna, %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 39:    // MayorIgual
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"MayorIgual,    %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 40:    // *MayorQue
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"MayorQue,      %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;
```

```
case 42:    // *CorrimientoDer
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"CorrimientoDer,      %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 43:    // CorrimientoDerAsigna
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"CorrimientoDerAsigna, %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 45:    // PotenciaAsigna
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"PotenciaAsigna,      %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 46:    // *Potencia
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Potencia,            %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 49:    // *Or
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Or,            %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 50:    // *Pipe
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Pipe,          %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 53:    // *Ampersand
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Ampersand,     %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 54:    // *And
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"SumaAsigna,    %s\n",arre);
```

```
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 56:    // *CorcheteIzq
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"CorcheteIzq,   %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 58:    // *CorcheteDer
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"CorcheteDer,   %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 60:    // *ParentesisIzq
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"ParentesisIzq, %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 62:    // *ParentesisDer
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"ParentesisDer, %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 64:    // *LlaveIzq
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"LlaveIzq,      %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 66:    // *LlaveDer
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"LlaveDer,      %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 69:    // *Texto
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Texto,         %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;
```

```
case 71:    // *PuntoYcoma
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"PuntoYcoma,    %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 73:    // *Coma
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Coma,          %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 75:    // Gatito
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Gatito,        %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 77:    // *Espacios
     edo=1;
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     ban2=1;
     break;

case 79:    // GuionBajo
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"GuionBajo,     %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 81:    // ComillaSimple
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"ComillaSimple, %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 83:    // *Punto
     edo=1;
     corrige_arreglo(arre);
     fprintf(apt2,"Punto,         %s\n",arre);
     strncpy(arre,"",100);
     ind=0;
     ban=1;
     break;

case 85:    // *Identificador
     edo=1;
     yesp=0;
     if (ban2==1) corrige_arreglo(arre);
     for(int b=0; b<75; b++){
        ptr = strcmp(arre,palabras[b]);
```

```
            if(ptr == 0){
             fprintf(apt2,"PalReservada,   %s\n",arre);
             yesp = 1;
             }
        }
        if(yesp == 0)
          fprintf(apt2,"Identificador,    %s\n",arre);
          strncpy(arre,"",100);
          ban=1;
          ind=0;
        break;

  case 86:    // *Objeto
        edo=1;
        corrige_arreglo(arre);
        fprintf(apt2,"Objeto,        %s\n",arre);
        strncpy(arre,"",100);
        ind=0;
        ban=1;
        break;

  case 90:    // *NumeroReal
        edo=1;
        corrige_arreglo(arre);
        fprintf(apt2,"NumeroReal,    %s\n",arre);
        strncpy(arre,"",100);
        ind=0;
        ban=1;
        break;

  case 91:    // *NumeroEntero
        edo=1;
        corrige_arreglo(arre);
        fprintf(apt2,"NumeroEntero,  %s\n",arre);
        strncpy(arre,"",100);
        ind=0;
        ban=1;
        break;

  case 93:    // NuevaLinea
        edo=1;
        strncpy(arre,"",100);
        ind=0;
        ban=1;
        break;

  case 95:    // *DosPuntos
        edo=1;
        corrige_arreglo(arre);
        fprintf(apt2,"DosPuntos,     %s\n",arre);
        strncpy(arre,"",100);
        ind=0;
        ban=1;
        break;
 } //fin Switch

 if(arre[0]==' '){
    ind=0; ban=0;
   }
 else
   arre[ind]=c;
 ind++;
} //fin while

fclose(apt2);
fclose(apt);
printf("Se ha creado un archivo con los tokens!!\n Presione enter para continuar");
getch();
```

```
exit(0);
}

void corrige_arreglo(char arre[100]){
  for(int r=1; r<=100; r++)
  arre[r-1]=arre[r];
}

void llenar_matriz(){
 // Llenado Horizontal de la fila[0] columna [e] con 0's
  for(int e=0; e<=30; e++)
    MT[0][e]=0;
 // Llenado Vertical de la fila[m] columna[0] con 0's
  for(int m=0; m==93; m++)
    MT[m][0]=0;
 // Transiciones del automata!!
MT[1][1]= 2;   MT[2][1]= 5;  MT[6][1]= 8;  MT[10][1]= 12;  MT[14][1]= 15;
MT[1][2]= 6;   MT[2][2]= 4;  MT[6][2]= 9;  MT[10][2]= 12;  MT[14][2]= 15;
MT[1][3]= 10;  MT[2][3]= 4;  MT[6][3]= 8;  MT[10][3]= 12;  MT[14][3]= 19;
MT[1][4]= 14;  MT[2][4]= 4;  MT[6][4]= 8;  MT[10][4]= 13;  MT[14][4]= 17;
MT[1][5]= 82;  MT[2][5]= 4;  MT[6][5]= 8;  MT[10][5]= 12;  MT[14][5]= 15;
MT[1][6]= 84;  MT[2][6]= 4;  MT[6][6]= 8;  MT[10][6]= 12;  MT[14][6]= 15;
MT[1][7]= 23;  MT[2][7]= 3;  MT[6][7]=11;  MT[10][7]= 11;  MT[14][7]= 16;
MT[1][8]= 26;  MT[2][8]= 4;  MT[6][8]= 8;  MT[10][8]= 12;  MT[14][8]= 15;
MT[1][9]= 29;  MT[2][9]= 4;  MT[6][9]= 8;  MT[10][9]= 12;  MT[14][9]= 15;
MT[1][10]=32;  MT[2][10]=4;  MT[6][10]=8;  MT[10][10]=12;  MT[14][10]=15;
MT[1][11]=38;  MT[2][11]=4;  MT[6][11]=8;  MT[10][11]=12;  MT[14][11]=15;
MT[1][12]=44;  MT[2][12]=4;  MT[6][12]=8;  MT[10][12]=12;  MT[14][12]=15;
MT[1][13]=47;  MT[2][13]=4;  MT[6][13]=8;  MT[10][13]=12;  MT[14][13]=15;
MT[1][14]=51;  MT[2][14]=4;  MT[6][14]=8;  MT[10][14]=12;  MT[14][14]=15;
MT[1][15]=55;  MT[2][15]=4;  MT[6][15]=8;  MT[10][15]=12;  MT[14][15]=15;
MT[1][16]=57;  MT[2][16]=4;  MT[6][16]=8;  MT[10][16]=12;  MT[14][16]=15;
MT[1][17]=59;  MT[2][17]=4;  MT[6][17]=8;  MT[10][17]=12;  MT[14][17]=15;
MT[1][18]=61;  MT[2][18]=4;  MT[6][18]=8;  MT[10][18]=12;  MT[14][18]=15;
MT[1][19]=63;  MT[2][19]=4;  MT[6][19]=8;  MT[10][19]=12;  MT[14][19]=15;
MT[1][20]=65;  MT[2][20]=4;  MT[6][20]=8;  MT[10][20]=12;  MT[14][20]=15;
MT[1][21]=67;  MT[2][21]=4;  MT[6][21]=8;  MT[10][21]=12;  MT[14][21]=15;
MT[1][22]=70;  MT[2][22]=4;  MT[6][22]=8;  MT[10][22]=12;  MT[14][22]=15;
MT[1][23]=72;  MT[2][23]=4;  MT[6][23]=8;  MT[10][23]=12;  MT[14][23]=15;
MT[1][24]=74;  MT[2][24]=4;  MT[6][24]=8;  MT[10][24]=12;  MT[14][24]=15;
MT[1][25]=76;  MT[2][25]=4;  MT[6][25]=8;  MT[10][25]=12;  MT[14][25]=15;
MT[1][26]=78;  MT[2][26]=4;  MT[6][26]=8;  MT[10][26]=12;  MT[14][26]=15;
MT[1][27]=80;  MT[2][27]=4;  MT[6][27]=8;  MT[10][27]=12;  MT[14][27]=15;
MT[1][28]=87;  MT[2][28]=4;  MT[6][28]=8;  MT[10][28]=12;  MT[14][28]=15;
MT[1][29]=92;  MT[2][29]=4;  MT[6][29]=8;  MT[10][29]=12;  MT[14][29]=15;
MT[1][30]=94;  MT[2][30]=4;  MT[6][30]=8;  MT[10][30]=12;  MT[14][30]=15;

MT[17][1]= 17;  MT[19][1]= 19;  MT[20][1]= 19;  MT[23][1]= 25;  MT[26][1]= 28;
MT[17][2]= 17;  MT[19][2]= 19;  MT[20][2]= 19;  MT[23][2]= 25;  MT[26][2]= 28;
MT[17][3]= 17;  MT[19][3]= 20;  MT[20][3]= 19;  MT[23][3]= 25;  MT[26][3]= 28;
MT[17][4]= 17;  MT[19][4]= 19;  MT[20][4]= 21;  MT[23][4]= 25;  MT[26][4]= 28;
MT[17][5]= 17;  MT[19][5]= 19;  MT[20][5]= 19;  MT[23][5]= 25;  MT[26][5]= 28;
MT[17][6]= 17;  MT[19][6]= 19;  MT[20][6]= 19;  MT[23][6]= 25;  MT[26][6]= 28;
MT[17][7]= 17;  MT[19][7]= 19;  MT[20][7]= 19;  MT[23][7]= 24;  MT[26][7]= 27;
MT[17][8]= 17;  MT[19][8]= 19;  MT[20][8]= 19;  MT[23][8]= 25;  MT[26][8]= 28;
MT[17][9]= 17;  MT[19][9]= 19;  MT[20][9]= 19;  MT[23][9]= 25;  MT[26][9]= 28;
MT[17][10]=17;  MT[19][10]=19;  MT[20][10]=19;  MT[23][10]=25;  MT[26][10]=28;
MT[17][11]=17;  MT[19][11]=19;  MT[20][11]=19;  MT[23][11]=25;  MT[26][11]=28;
MT[17][12]=17;  MT[19][12]=19;  MT[20][12]=19;  MT[23][12]=25;  MT[26][12]=28;
MT[17][13]=17;  MT[19][13]=19;  MT[20][13]=19;  MT[23][13]=25;  MT[26][13]=28;
MT[17][14]=17;  MT[19][14]=19;  MT[20][14]=19;  MT[23][14]=25;  MT[26][14]=28;
MT[17][15]=17;  MT[19][15]=19;  MT[20][15]=19;  MT[23][15]=25;  MT[26][15]=28;
MT[17][16]=17;  MT[19][16]=19;  MT[20][16]=19;  MT[23][16]=25;  MT[26][16]=28;
MT[17][17]=17;  MT[19][17]=19;  MT[20][17]=19;  MT[23][17]=25;  MT[26][17]=28;
MT[17][18]=17;  MT[19][18]=19;  MT[20][18]=19;  MT[23][18]=25;  MT[26][18]=28;
MT[17][19]=17;  MT[19][19]=19;  MT[20][19]=19;  MT[23][19]=25;  MT[26][19]=28;
MT[17][20]=17;  MT[19][20]=19;  MT[20][20]=19;  MT[23][20]=25;  MT[26][20]=28;
MT[17][21]=17;  MT[19][21]=19;  MT[20][21]=19;  MT[23][21]=25;  MT[26][21]=28;
```

```
MT[17][22]=17;  MT[19][22]=19;  MT[20][22]=19;  MT[23][22]=25;  MT[26][22]=28;
MT[17][23]=17;  MT[19][23]=19;  MT[20][23]=19;  MT[23][23]=25;  MT[26][23]=28;
MT[17][24]=17;  MT[19][24]=19;  MT[20][24]=19;  MT[23][24]=25;  MT[26][24]=28;
MT[17][25]=17;  MT[19][25]=19;  MT[20][25]=19;  MT[23][25]=25;  MT[26][25]=28;
MT[17][26]=17;  MT[19][26]=19;  MT[20][26]=19;  MT[23][26]=25;  MT[26][26]=28;
MT[17][27]=17;  MT[19][27]=19;  MT[20][27]=19;  MT[23][27]=25;  MT[26][27]=28;
MT[17][28]=17;  MT[19][28]=19;  MT[20][28]=19;  MT[23][28]=25;  MT[26][28]=28;
MT[17][29]=18;  MT[19][29]=19;  MT[20][29]=19;  MT[23][29]=25;  MT[26][29]=28;
MT[17][30]=18;  MT[19][30]=19;  MT[20][30]=19;  MT[23][30]=25;  MT[26][30]=28;

MT[29][1]= 31;  MT[32][1]= 34;  MT[35][1]= 36;  MT[38][1]= 40;  MT[41][1]= 42;
MT[29][2]= 31;  MT[32][2]= 34;  MT[35][2]= 36;  MT[38][2]= 40;  MT[41][2]= 42;
MT[29][3]= 31;  MT[32][3]= 34;  MT[35][3]= 36;  MT[38][3]= 40;  MT[41][3]= 42;
MT[29][4]= 31;  MT[32][4]= 34;  MT[35][4]= 36;  MT[38][4]= 40;  MT[41][4]= 42;
MT[29][5]= 31;  MT[32][5]= 34;  MT[35][5]= 36;  MT[38][5]= 40;  MT[41][5]= 42;
MT[29][6]= 31;  MT[32][6]= 34;  MT[35][6]= 36;  MT[38][6]= 40;  MT[41][6]= 42;
MT[29][7]= 30;  MT[32][7]= 33;  MT[35][7]= 37;  MT[38][7]= 39;  MT[41][7]= 43;
MT[29][8]= 31;  MT[32][8]= 34;  MT[35][8]= 36;  MT[38][8]= 40;  MT[41][8]= 42;
MT[29][9]= 31;  MT[32][9]= 34;  MT[35][9]= 36;  MT[38][9]= 40;  MT[41][9]= 42;
MT[29][10]=31;  MT[32][10]=35;  MT[35][10]=36;  MT[38][10]=40;  MT[41][10]=42;
MT[29][11]=31;  MT[32][11]=34;  MT[35][11]=36;  MT[38][11]=41;  MT[41][11]=42;
MT[29][12]=31;  MT[32][12]=34;  MT[35][12]=36;  MT[38][12]=40;  MT[41][12]=42;
MT[29][13]=31;  MT[32][13]=34;  MT[35][13]=36;  MT[38][13]=40;  MT[41][13]=42;
MT[29][14]=31;  MT[32][14]=34;  MT[35][14]=36;  MT[38][14]=40;  MT[41][14]=42;
MT[29][15]=31;  MT[32][15]=34;  MT[35][15]=36;  MT[38][15]=40;  MT[41][15]=42;
MT[29][16]=31;  MT[32][16]=34;  MT[35][16]=36;  MT[38][16]=40;  MT[41][16]=42;
MT[29][17]=31;  MT[32][17]=34;  MT[35][17]=36;  MT[38][17]=40;  MT[41][17]=42;
MT[29][18]=31;  MT[32][18]=34;  MT[35][18]=36;  MT[38][18]=40;  MT[41][18]=42;
MT[29][19]=31;  MT[32][19]=34;  MT[35][19]=36;  MT[38][19]=40;  MT[41][19]=42;
MT[29][20]=31;  MT[32][20]=34;  MT[35][20]=36;  MT[38][20]=40;  MT[41][20]=42;
MT[29][21]=31;  MT[32][21]=34;  MT[35][21]=36;  MT[38][21]=40;  MT[41][21]=42;
MT[29][22]=31;  MT[32][22]=34;  MT[35][22]=36;  MT[38][22]=40;  MT[41][22]=42;
MT[29][23]=31;  MT[32][23]=34;  MT[35][23]=36;  MT[38][23]=40;  MT[41][23]=42;
MT[29][24]=31;  MT[32][24]=34;  MT[35][24]=36;  MT[38][24]=40;  MT[41][24]=42;
MT[29][25]=31;  MT[32][25]=34;  MT[35][25]=36;  MT[38][25]=40;  MT[41][25]=42;
MT[29][26]=31;  MT[32][26]=34;  MT[35][26]=36;  MT[38][26]=40;  MT[41][26]=42;
MT[29][27]=31;  MT[32][27]=34;  MT[35][27]=36;  MT[38][27]=40;  MT[41][27]=42;
MT[29][28]=31;  MT[32][28]=34;  MT[35][28]=36;  MT[38][28]=40;  MT[41][28]=42;
MT[29][29]=31;  MT[32][29]=34;  MT[35][29]=36;  MT[38][29]=40;  MT[41][29]=42;
MT[29][30]=31;  MT[32][30]=34;  MT[35][30]=36;  MT[38][30]=40;  MT[41][30]=42;

MT[44][1]= 46;  MT[47][1]= 49;  MT[51][1]= 53;  MT[67][1]= 67;  MT[76][1]= 77;
MT[44][2]= 46;  MT[47][2]= 49;  MT[51][2]= 53;  MT[67][2]= 67;  MT[76][2]= 77;
MT[44][3]= 46;  MT[47][3]= 49;  MT[51][3]= 53;  MT[67][3]= 67;  MT[76][3]= 77;
MT[44][4]= 46;  MT[47][4]= 49;  MT[51][4]= 53;  MT[67][4]= 67;  MT[76][4]= 77;
MT[44][5]= 46;  MT[47][5]= 49;  MT[51][5]= 53;  MT[67][5]= 67;  MT[76][5]= 77;
MT[44][6]= 46;  MT[47][6]= 49;  MT[51][6]= 53;  MT[67][6]= 67;  MT[76][6]= 77;
MT[44][7]= 45;  MT[47][7]= 49;  MT[51][7]= 53;  MT[67][7]= 67;  MT[76][7]= 77;
MT[44][8]= 46;  MT[47][8]= 49;  MT[51][8]= 53;  MT[67][8]= 67;  MT[76][8]= 77;
MT[44][9]= 46;  MT[47][9]= 49;  MT[51][9]= 53;  MT[67][9]= 67;  MT[76][9]= 77;
MT[44][10]=46;  MT[47][10]=49;  MT[51][10]=53;  MT[67][10]=67;  MT[76][10]=77;
MT[44][11]=46;  MT[47][11]=49;  MT[51][11]=53;  MT[67][11]=67;  MT[76][11]=77;
MT[44][12]=46;  MT[47][12]=49;  MT[51][12]=53;  MT[67][12]=67;  MT[76][12]=77;
MT[44][13]=46;  MT[47][13]=48;  MT[51][13]=53;  MT[67][13]=67;  MT[76][13]=77;
MT[44][14]=46;  MT[47][14]=49;  MT[51][14]=52;  MT[67][14]=67;  MT[76][14]=77;
MT[44][15]=46;  MT[47][15]=49;  MT[51][15]=53;  MT[67][15]=67;  MT[76][15]=77;
MT[44][16]=46;  MT[47][16]=49;  MT[51][16]=53;  MT[67][16]=67;  MT[76][16]=77;
MT[44][17]=46;  MT[47][17]=49;  MT[51][17]=53;  MT[67][17]=67;  MT[76][17]=77;
MT[44][18]=46;  MT[47][18]=49;  MT[51][18]=53;  MT[67][18]=67;  MT[76][18]=77;
MT[44][19]=46;  MT[47][19]=49;  MT[51][19]=53;  MT[67][19]=67;  MT[76][19]=77;
MT[44][20]=46;  MT[47][20]=49;  MT[51][20]=53;  MT[67][20]=67;  MT[76][20]=77;
MT[44][21]=46;  MT[47][21]=49;  MT[51][21]=53;  MT[67][21]=68;  MT[76][21]=77;
MT[44][22]=46;  MT[47][22]=49;  MT[51][22]=53;  MT[67][22]=67;  MT[76][22]=77;
MT[44][23]=46;  MT[47][23]=49;  MT[51][23]=53;  MT[67][23]=67;  MT[76][23]=77;
MT[44][24]=46;  MT[47][24]=49;  MT[51][24]=53;  MT[67][24]=67;  MT[76][24]=77;
MT[44][25]=46;  MT[47][25]=49;  MT[51][25]=53;  MT[67][25]=67;  MT[76][25]=76;
MT[44][26]=46;  MT[47][26]=49;  MT[51][26]=53;  MT[67][26]=67;  MT[76][26]=77;
MT[44][27]=46;  MT[47][27]=49;  MT[51][27]=53;  MT[67][27]=67;  MT[76][27]=77;
```

```
MT[44][28]=46;  MT[47][28]=49;  MT[51][28]=53;  MT[67][28]=67;  MT[76][28]=77;
MT[44][29]=46;  MT[47][29]=49;  MT[51][29]=53;  MT[67][29]=67;  MT[76][29]=77;
MT[44][30]=46;  MT[47][30]=49;  MT[51][30]=53;  MT[67][30]=67;  MT[76][30]=77;

MT[78][1]= 79;  MT[82][1]= 83;  MT[84][1]= 85;  MT[87][1]= 91;  MT[88][1]= 90;
MT[78][2]= 79;  MT[82][2]= 83;  MT[84][2]= 85;  MT[87][2]= 91;  MT[88][2]= 90;
MT[78][3]= 79;  MT[82][3]= 83;  MT[84][3]= 85;  MT[87][3]= 91;  MT[88][3]= 90;
MT[78][4]= 79;  MT[82][4]= 83;  MT[84][4]= 85;  MT[87][4]= 91;  MT[88][4]= 90;
MT[78][5]= 79;  MT[82][5]= 83;  MT[84][5]= 86;  MT[87][5]= 88;  MT[88][5]= 90;
MT[78][6]= 78;  MT[82][6]= 83;  MT[84][6]= 84;  MT[87][6]= 91;  MT[88][6]= 90;
MT[78][7]= 79;  MT[82][7]= 83;  MT[84][7]= 85;  MT[87][7]= 91;  MT[88][7]= 90;
MT[78][8]= 79;  MT[82][8]= 83;  MT[84][8]= 85;  MT[87][8]= 91;  MT[88][8]= 90;
MT[78][9]= 79;  MT[82][9]= 83;  MT[84][9]= 85;  MT[87][9]= 91;  MT[88][9]= 90;
MT[78][10]=79;  MT[82][10]=83;  MT[84][10]=85;  MT[87][10]=91;  MT[88][10]=90;
MT[78][11]=79;  MT[82][11]=83;  MT[84][11]=85;  MT[87][11]=91;  MT[88][11]=90;
MT[78][12]=79;  MT[82][12]=83;  MT[84][12]=85;  MT[87][12]=91;  MT[88][12]=90;
MT[78][13]=79;  MT[82][13]=83;  MT[84][13]=85;  MT[87][13]=91;  MT[88][13]=90;
MT[78][14]=79;  MT[82][14]=83;  MT[84][14]=85;  MT[87][14]=91;  MT[88][14]=90;
MT[78][15]=79;  MT[82][15]=83;  MT[84][15]=85;  MT[87][15]=91;  MT[88][15]=90;
MT[78][16]=79;  MT[82][16]=83;  MT[84][16]=85;  MT[87][16]=91;  MT[88][16]=90;
MT[78][17]=79;  MT[82][17]=83;  MT[84][17]=85;  MT[87][17]=91;  MT[88][17]=90;
MT[78][18]=79;  MT[82][18]=83;  MT[84][18]=85;  MT[87][18]=91;  MT[88][18]=90;
MT[78][19]=79;  MT[82][19]=83;  MT[84][19]=85;  MT[87][19]=91;  MT[88][19]=90;
MT[78][20]=79;  MT[82][20]=83;  MT[84][20]=85;  MT[87][20]=91;  MT[88][20]=90;
MT[78][21]=79;  MT[82][21]=83;  MT[84][21]=85;  MT[87][21]=91;  MT[88][21]=90;
MT[78][22]=79;  MT[82][22]=83;  MT[84][22]=85;  MT[87][22]=91;  MT[88][22]=90;
MT[78][23]=79;  MT[82][23]=83;  MT[84][23]=85;  MT[87][23]=91;  MT[88][23]=90;
MT[78][24]=79;  MT[82][24]=83;  MT[84][24]=85;  MT[87][24]=91;  MT[88][24]=90;
MT[78][25]=79;  MT[82][25]=83;  MT[84][25]=85;  MT[87][25]=91;  MT[88][25]=90;
MT[78][26]=78;  MT[82][26]=83;  MT[84][26]=85;  MT[87][26]=91;  MT[88][26]=90;
MT[78][27]=79;  MT[82][27]=83;  MT[84][27]=85;  MT[87][27]=91;  MT[88][27]=90;
MT[78][28]=78;  MT[82][28]=83;  MT[84][28]=84;  MT[87][28]=87;  MT[88][28]=89;
MT[78][29]=79;  MT[82][29]=83;  MT[84][29]=85;  MT[87][29]=91;  MT[88][29]=90;
MT[78][30]=79;  MT[82][30]=83;  MT[84][30]=85;  MT[87][30]=91;  MT[88][30]=90;

MT[89][1]= 90;  MT[92][1]= 93;
MT[89][2]= 90;  MT[92][2]= 93;
MT[89][3]= 90;  MT[92][3]= 93;
MT[89][4]= 90;  MT[92][4]= 93;
MT[89][5]= 90;  MT[92][5]= 93;
MT[89][6]= 90;  MT[92][6]= 93;
MT[89][7]= 90;  MT[92][7]= 93;
MT[89][8]= 90;  MT[92][8]= 93;
MT[89][9]= 90;  MT[92][9]= 93;
MT[89][10]=90;  MT[92][10]=93;
MT[89][11]=90;  MT[92][11]=93;
MT[89][12]=90;  MT[92][12]=93;
MT[89][13]=90;  MT[92][13]=93;
MT[89][14]=90;  MT[92][14]=93;
MT[89][15]=90;  MT[92][15]=93;
MT[89][16]=90;  MT[92][16]=93;
MT[89][17]=90;  MT[92][17]=93;
MT[89][18]=90;  MT[92][18]=93;
MT[89][19]=90;  MT[92][19]=93;
MT[89][20]=90;  MT[92][20]=93;
MT[89][21]=90;  MT[92][21]=93;
MT[89][22]=90;  MT[92][22]=93;
MT[89][23]=90;  MT[92][23]=93;
MT[89][24]=90;  MT[92][24]=93;
MT[89][25]=90;  MT[92][25]=92;
MT[89][26]=90;  MT[92][26]=93;
MT[89][27]=90;  MT[92][27]=93;
MT[89][28]=89;  MT[92][28]=93;
MT[89][29]=90;  MT[92][29]=92;
MT[89][30]=90;  MT[92][30]=93;

// Columnas con valores repetitivos
 for(int i=1; i<=30; i++){
```

```
  MT[21][i]=22;  MT[48][i]=50;  MT[52][i]=54;  MT[55][i]=56;  MT[57][i]=58;
  MT[59][i]=60;  MT[61][i]=62;  MT[63][i]=64;  MT[65][i]=66;  MT[68][i]=69;
  MT[70][i]=71;  MT[72][i]=73;  MT[74][i]=75;  MT[80][i]=81;  MT[82][i]=83;
  MT[94][i]=95;
 }
}//fin de la funcion llenar_matriz()
```