

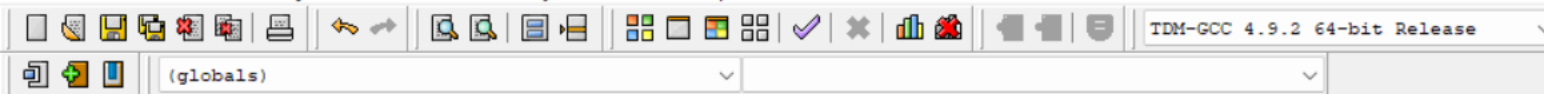
Programa 1

Mini compilador (usando pila)

```
1  #include<stdio.h>
2  #include<conio.h>
3  #include<stdlib.h>
4  #include<string.h>
5  #include "pila.c"
6
7  char nombre[10] ;
8  FILE *fichero;
9  int main(int argc, char *argv[]) {
10
11     int i=0 ;
12     int n;
13     char formula[100], archivo[100];
14     printf("\t\t prac 1.c\n");
15     printf("1.  Archivo 1.c\n");
16     printf("2.  Archivo 2.c\n");
17     printf("3.  Archivo 3.c\n");
18     printf("4.  Archivo 4.c\n");
19     printf("5.  Salir\n");
20     printf("Dame el numero  de arcivo: ");
21     scanf("%d", &n);
22     switch(n) {
23     case 1:{
24         strcpy(archivo,"1.c");
25         break; }
26     case 2:{
27         strcpy(archivo,"2.c");
```

practica 1

Line	Col	File	Message



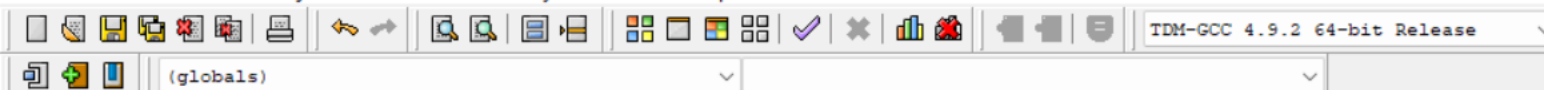
Project Classe [*] practica1.c

```
19 printf("5. Salir\n");
20 printf("Dame el numero de archivo: ");
21 scanf("%d", &n);
22 switch(n) {
23 case 1:{
24     strcpy(archivo,"1.c");
25     break; }
26 case 2:{
27     strcpy(archivo,"2.c");
28     break; }
29 case 3:{
30     strcpy(archivo,"3.c");
31     break;
32 }
33 case 4:{
34     strcpy(archivo,"4.c");
35     break;
36 }
37 default:{
38     puts("Error");
39     break;
40 }
41 }
42
43
44 fichero = fopen(archivo, "r" );
45 printf("Fichero %s\n", fichero);
```

Compiler Resources Compile Log Debug Find Results Close

Line	Col	File	Message

Line: 132 Col: 4 Sel: 654 Lines: 155 Length: 3892 Insert Done parsing in 0.266 seconds

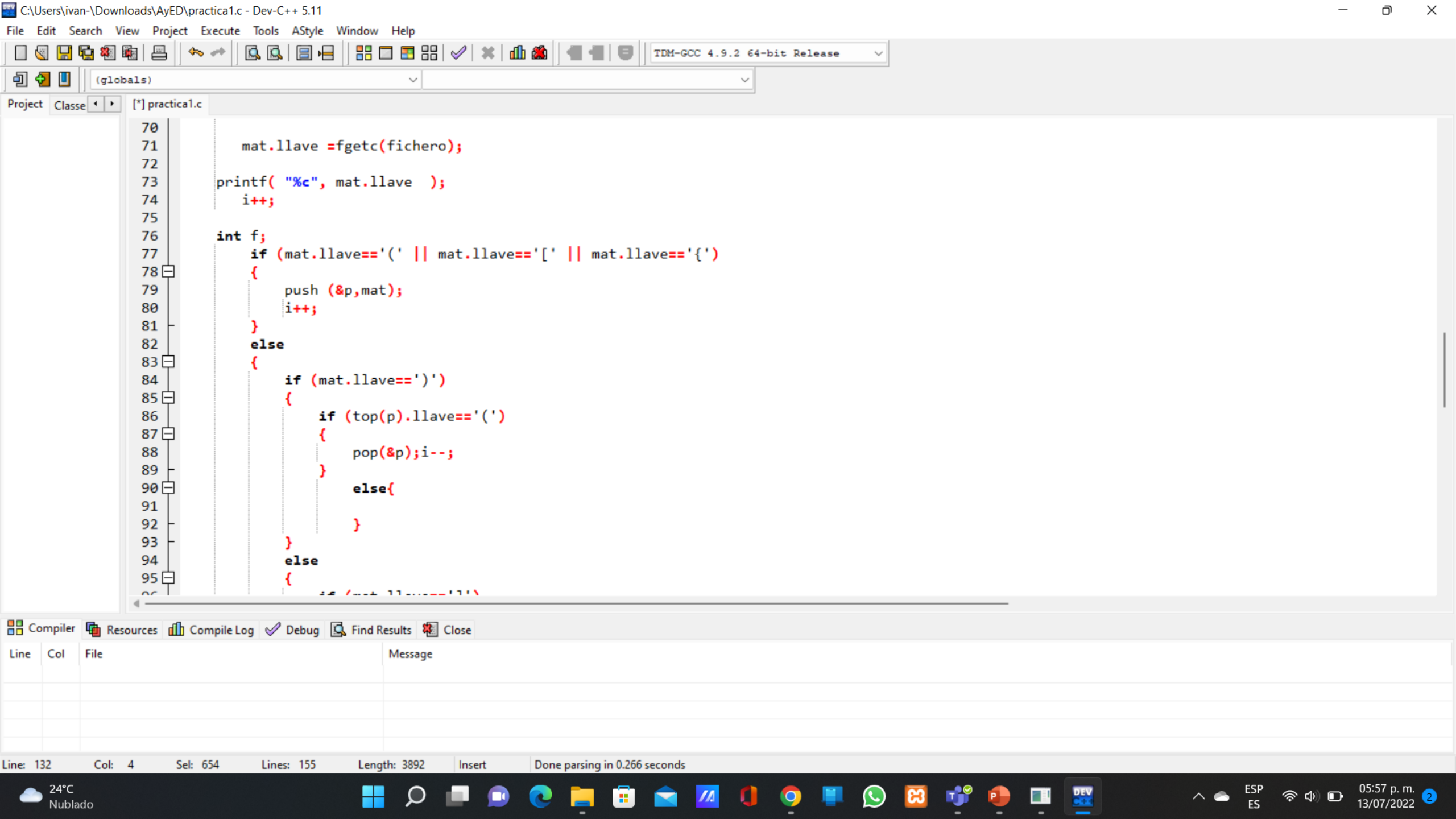


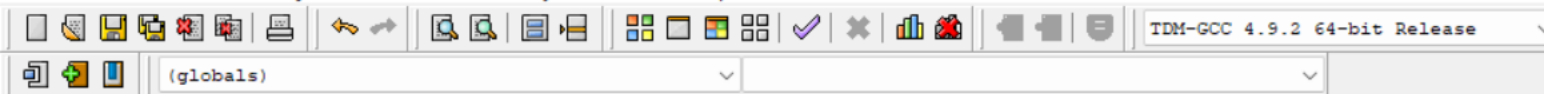
```
Project Classe [*] practica1.c
40 }
41 }
42
43
44 fichero = fopen(archivo, "r" );
45 printf( "Fichero: %s -> ", archivo );
46 if( fichero )
47     printf( "existe (ABIERTO)\n" );
48 else
49 {
50     printf( "Error (NO ABIERTO)\n" );
51     return 1;
52 }
53
54 // char mat[100];
55 Info mat ;
56 mat.columna= 1 ;
57 mat.linea= 1 ;
58 pila p;
59 crearpila (&p);
60 i=-76 ;
61 printf( "Los 18 primeros caracteres del fichero: %s\n\n", nombre );
62 while(mat.llave!= EOF){
63     if (mat.llave=='\n')
64     {
65         mat.columna= 1 ;
66     }
```

Compiler Resources Compile Log Debug Find Results Close

Line	Col	File	Message

Line: 132 Col: 4 Sel: 654 Lines: 155 Length: 3892 Insert Done parsing in 0.266 seconds





Project Classe (*) practical.c

```
100         pop(&p);i--;
101     }
102     else{
103     }
104 }
105 }
106 else
107 {
108     if (mat.llave=='}')
109     {
110     }
111     if (empty (p))
112     {
113         if (mat.llave=='{' || mat.llave==']' || mat.llave=='}')
114         {
115             printf("La formula no esta correctamente balanceada\n");
116             printf("Error tipo 2 hace falta (, [ o { que abre antes de la linea %d columna %d ",top(p).linea,top(p).columna);
117             return 1;
118         }
119     }
120     else
121     {
122         if (top(p).llave=='{' )
123         {
124             pop(&p);i--;
125         }
126     }
```

Compiler Resources Compile Log Debug Find Results Close

Line	Col	File	Message

Line: 132 Col: 4 Sel: 654 Lines: 155 Length: 3892 Insert Done parsing in 0.266 seconds

```
Project Classe [*] practica1.c
130     }
131 }
132
133     if (empty (p))
134     {
135         printf("La formula esta correctamente balanceada");
136     }
137     else
138     {
139
140         if (i>=2)
141         {
142             printf("La formula no esta correctamente balanceada\n");
143             printf("Error tipo 3  %c que esta en la línea %d columna %d no es del mismo tipo que %c de la línea %d columna %d ",pop(&p).llave,to
144             )else printf("La formula no esta correctamente balanceada\n");
145             printf("Error tipo 1 hace falta ), ] o } a partir de la línea %d columna %d ",top(p).linea,top(p).columna);
146         }
147     }
148     if( !fclose(fichero) )
149     printf( "\nFichero cerrado\n" );
150     else
151     {
152         printf( "\nError: fichero NO CERRADO\n" );
153         return 1;
154     }
155     return 0;
156 }
```

Line	Col	File	Message

```
prac 1.c
1. Archivo 1.c
2. Archivo 2.c
3. Archivo 3.c
4. Archivo 4.c
5. Salir
Dame el numero de archivo: 1.c
Fichero: 1.c -> existe (ABIERTO)
```

```
#include <stdio.h>
#include <stdlib.h>
```

```
void main ()
```

```
int a [6];
```

```
}La formula no esta correctamente balanceada
```

```
Error tipo 2 hace falta (, [ o { que abre antes de la linea 9 columna 2
```

```
-----
Process exited after 2.75 seconds with return value 1
```

```
Presione una tecla para continuar . . .
```



```
    prac 1.c
1.    Archivo 1.c
2.    Archivo 2.c
3.    Archivo 3.c
4.    Archivo 4.c
5.    Salir
Dame el numero de archivo: 2.c
Fichero: 2.c -> existe (ABIERTO)
Lr
```

```
#include <stdio.h>
#include <stdlib.h>
```

```
void main (){
```

```
    int a [6];
```

```
La formula no esta correctamente balanceada
Error tipo 1 hace falta ), ] o } a partir de la linea 5 columna 14
Fichero cerrado
```

```
-----
Process exited after 3.518 seconds with return value 0
Presione una tecla para continuar . . .
```

```
    prac 1.c
1.    Archivo 1.c
2.    Archivo 2.c
3.    Archivo 3.c
4.    Archivo 4.c
5.    Salir
Dame el numero de archivo: 3.c
Fichero: 3.c -> existe (ABIERTO)
```

```
#include <stdio.h>
#include <stdlib.h>
```

```
void main (){
```

```
    int a [6];
```

```
}
```

La formula no esta correctamente balanceada

Error tipo 3 { que esta en la lYnea 5 columna 14 no es del mismo tipo que [de la lYnea 8 columna 9

Fichero cerrado

Process exited after 3.425 seconds with return value 0

Presione una tecla para continuar . . .

```
prac 1.c
1. Archivo 1.c
2. Archivo 2.c
3. Archivo 3.c
4. Archivo 4.c
5. Salir
Dame el numero de archivo: 4.c
Fichero: 4.c -> existe (ABIERTO)
```

```
#include <stdio.h>
#include <stdlib.h>
```

```
void main (){
```

```
int a [6];
```

```
}
La formula esta correctamente balanceada
Fichero cerrado
```

```
-----
Process exited after 33.7 seconds with return value 0
Presione una tecla para continuar . . .
```