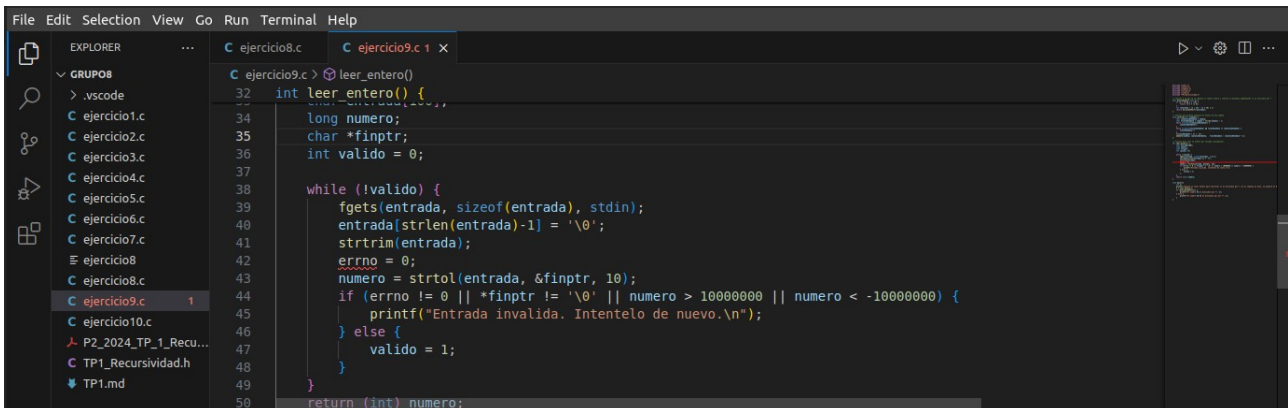


RESULTADO DE LA CORRECCIÓN: **APROBADO**

## OBSERVACIONES

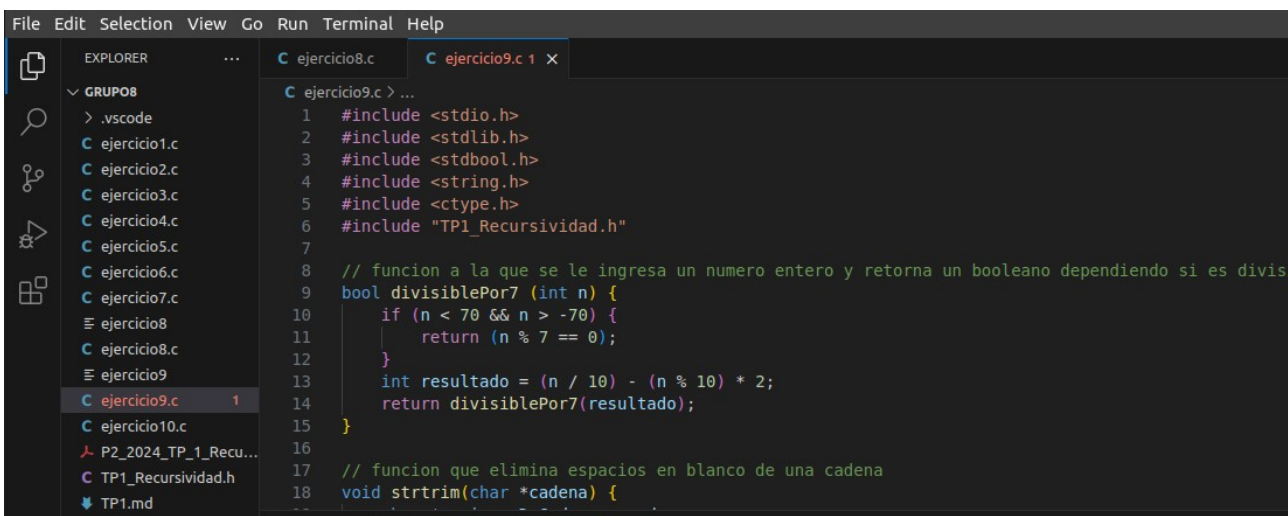
Crear un único archivo .c en donde se realicen las implementaciones de lo definido en el .h para facilitar los tests automáticos. Para funciones propias pueden crear sus propios .h y .c o sumarlas al .h subido a la plataforma.

En el ejercicio 9 faltaba realizar un include y fallaba la compilación. Por ser el primer TP no se reflejará en la nota. Muy buen control de carga de datos.



```

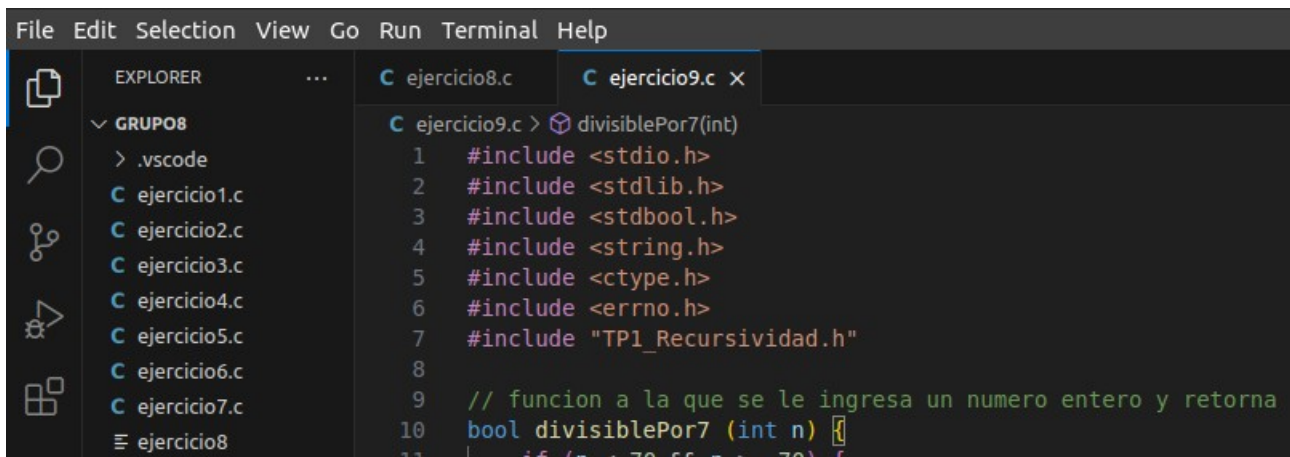
File Edit Selection View Go Run Terminal Help
EXPLORER
GRUPO8
  .vscode
  ejercicio1.c
  ejercicio2.c
  ejercicio3.c
  ejercicio4.c
  ejercicio5.c
  ejercicio6.c
  ejercicio7.c
  ejercicio8
  ejercicio8.c
  ejercicio9.c 1
  ejercicio10.c
  P2_2024_TP_1_Recu...
  TP1_Recursividad.h
  TP1.md
ejercicio9.c
32 int leer_entero() {
33     char entrada[1000000];
34     long numero;
35     char *finptr;
36     int valido = 0;
37
38     while (!valido) {
39         fgets(entrada, sizeof(entrada), stdin);
40         entrada[strlen(entrada)-1] = '\0';
41         strtrim(entrada);
42         errno = 0;
43         numero = strtol(entrada, &finptr, 10);
44         if (errno != 0 || *finptr != '\0' || numero > 10000000 || numero < -10000000) {
45             printf("Entrada invalida. Intentelo de nuevo.\n");
46         } else {
47             valido = 1;
48         }
49     }
50     return (int) numero;
  
```



```

File Edit Selection View Go Run Terminal Help
EXPLORER
GRUPO8
  .vscode
  ejercicio1.c
  ejercicio2.c
  ejercicio3.c
  ejercicio4.c
  ejercicio5.c
  ejercicio6.c
  ejercicio7.c
  ejercicio8
  ejercicio8.c
  ejercicio9
  ejercicio9.c 1
  ejercicio10.c
  P2_2024_TP_1_Recu...
  TP1_Recursividad.h
  TP1.md
ejercicio9.c
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <stdbool.h>
4 #include <string.h>
5 #include <ctype.h>
6 #include "TP1_Recursividad.h"
7
8 // funcion a la que se le ingresa un numero entero y retorna un booleano dependiendo si es divis
9 bool divisiblePor7 (int n) {
10     if (n < 70 && n > -70) {
11         return (n % 7 == 0);
12     }
13     int resultado = (n / 10) - (n % 10) * 2;
14     return divisiblePor7(resultado);
15 }
16
17 // funcion que elimina espacios en blanco de una cadena
18 void strtrim(char *cadena) {
  
```

## GRUPO 8 – Correcciones Trabajo Práctico: RECURSIVIDAD



The image shows a screenshot of the Visual Studio Code editor interface. The top menu bar includes 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', 'Terminal', and 'Help'. The left sidebar shows the 'EXPLORER' view with a file tree under the name 'GRUPO8'. The tree contains files from '.vscode' to 'ejercicio8'. The main editor area has two tabs open: 'ejercicio8.c' and 'ejercicio9.c'. The 'ejercicio9.c' tab is active, displaying the following C code:

```
C ejercicio9.c > divisiblePor7(int)
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <stdbool.h>
4  #include <string.h>
5  #include <ctype.h>
6  #include <errno.h>
7  #include "TP1_Recurividad.h"
8
9  // funcion a la que se le ingresa un numero entero y retorna
10 bool divisiblePor7 (int n) {
11     if (n < 70 && n >= 70) {
```