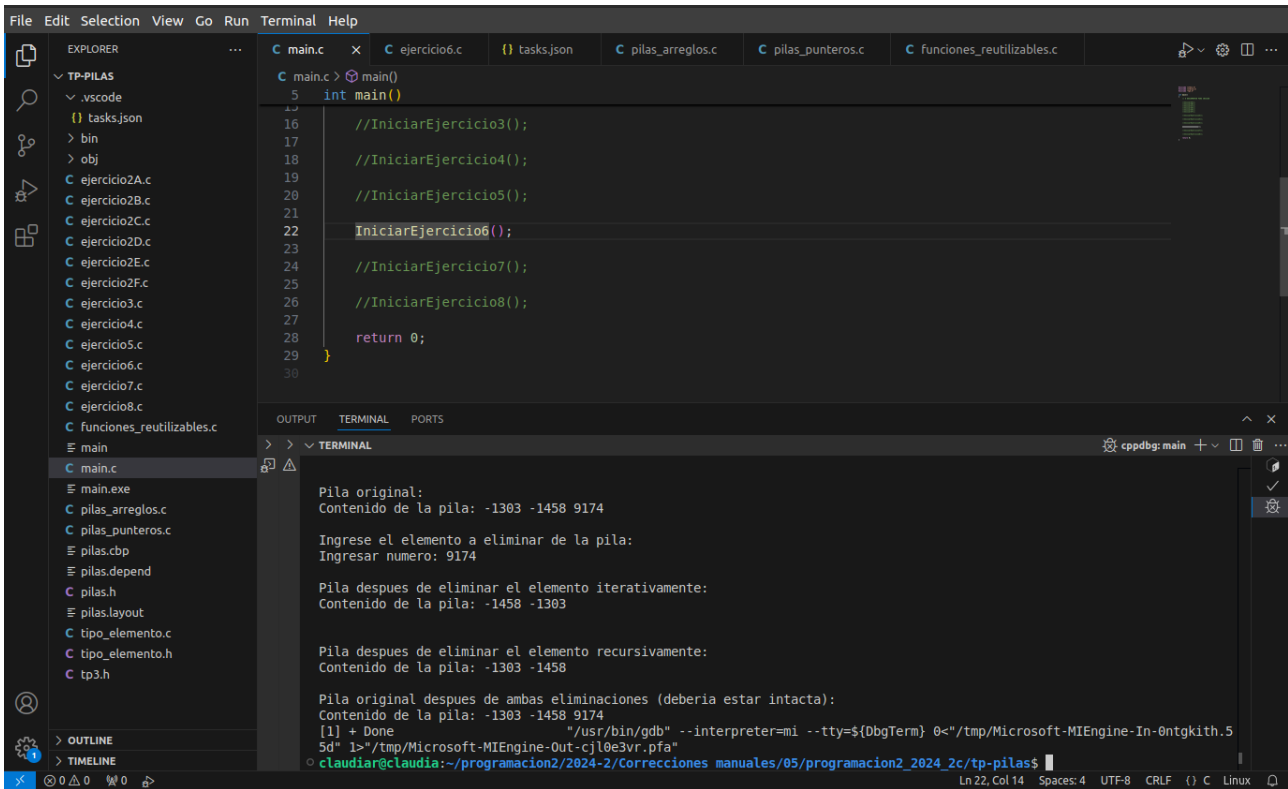


RESULTADO DE LA CORRECCIÓN: **APROBADO-**

OBSERVACIONES

No permite pila vacía. Controla el tamaño por código. No trabajan con el **return false** que el **p_apilar** devuelve si la pila está llena. En el ejercicio 6 no permite la carga manual de datos. En el ejercicio 8 no realiza ningún tipo de control de la cantidad de valores ingresados y luego solo procesa los válidos según el tamaño de la pila.



The screenshot shows a Visual Studio Code editor with a C project named 'TP-PILAS'. The Explorer sidebar on the left lists files including 'main.c', 'ejercicio2A.c', 'ejercicio2B.c', 'ejercicio2C.c', 'ejercicio2D.c', 'ejercicio2E.c', 'ejercicio2F.c', 'ejercicio3.c', 'ejercicio4.c', 'ejercicio5.c', 'ejercicio6.c', 'ejercicio7.c', 'ejercicio8.c', 'funciones_reutilizables.c', 'main.exe', 'pilas_arreglos.c', 'pilas_punteros.c', 'pilas.cbp', 'pilas.depend', 'pilas.h', 'pilas.layout', 'tipo_elemento.c', 'tipo_elemento.h', and 'tp3.h'. The main editor displays 'main.c' with the following code:

```
1 //main.c
2 #include <stdio.h>
3 #include "pilas.h"
4 #include "funciones_reutilizables.c"
5 int main()
6 {
7     //IniciarEjercicio3();
8     //IniciarEjercicio4();
9     //IniciarEjercicio5();
10    IniciarEjercicio6();
11    //IniciarEjercicio7();
12    //IniciarEjercicio8();
13    return 0;
14 }
```

The TERMINAL panel at the bottom shows the execution output:

```
cpdbg: main
Pila original:
Contenido de la pila: -1303 -1458 9174

Ingrese el elemento a eliminar de la pila:
Ingresar numero: 9174

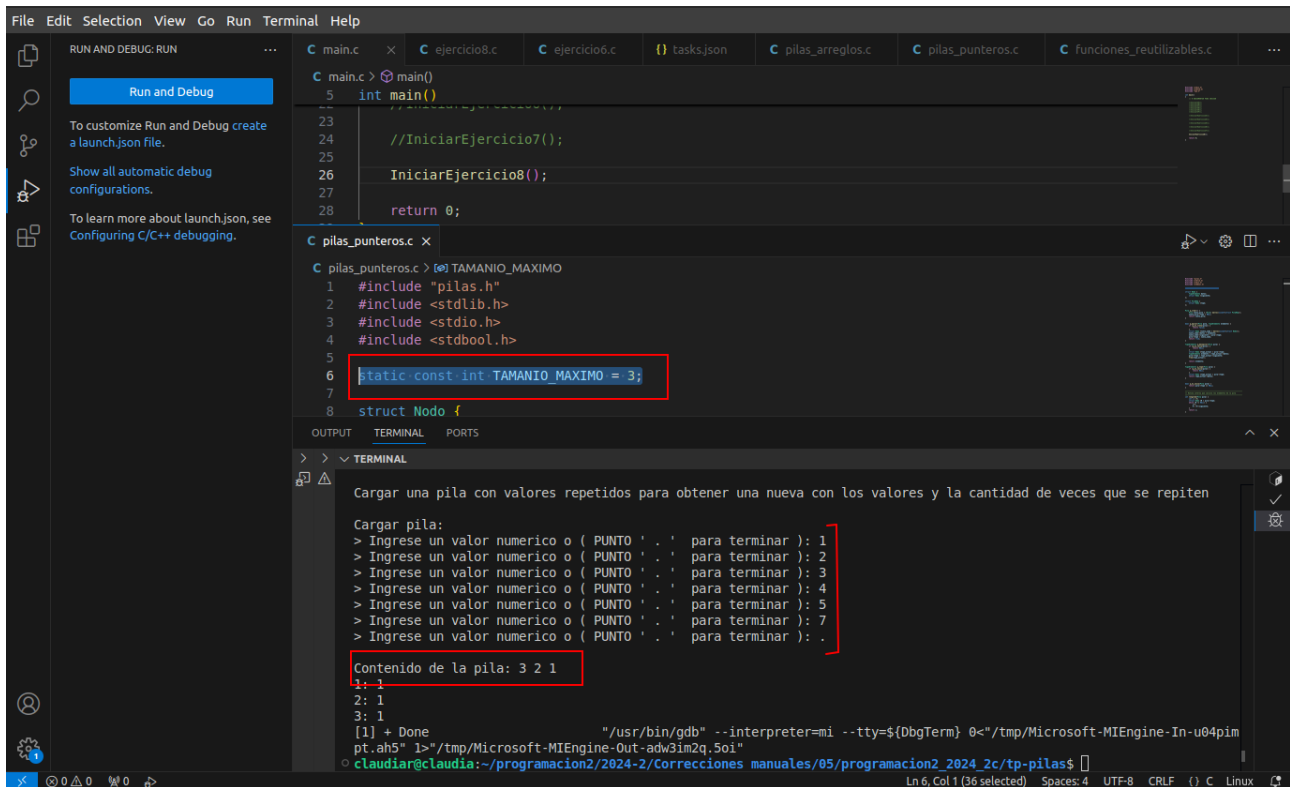
Pila despues de eliminar el elemento iterativamente:
Contenido de la pila: -1458 -1303

Pila despues de eliminar el elemento recursivamente:
Contenido de la pila: -1303 -1458

Pila original despues de ambas eliminaciones (deberia estar intacta):
Contenido de la pila: -1303 -1458 9174
[1] + Done
"/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm} 0<"/tmp/Microsoft-MIEngine-In-0ntgkith.5
5d" l>"/tmp/Microsoft-MIEngine-Out-cj10e3vr.pfa"
claudiar@claudia:~/programacion2/2024-2/Correcciones manuales/05/programacion2_2024_2c/tp-pilas$
```

The status bar at the bottom indicates 'Ln 22, Col 14', 'Spaces: 4', 'UTF-8', 'CRLF', and 'Linux'.

GRUPO 5 – Correcciones Trabajo Práctico: PILAS

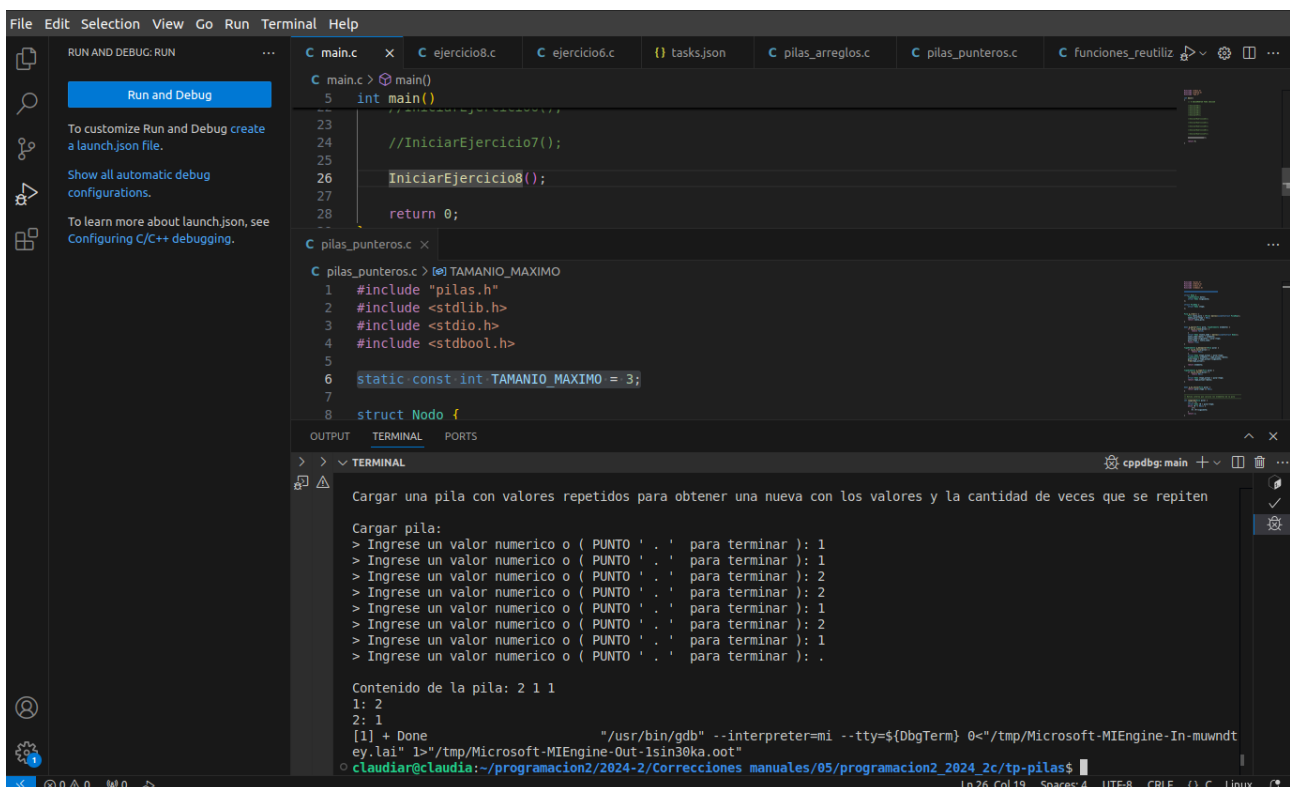


```
File Edit Selection View Go Run Terminal Help
RUN AND DEBUG: RUN
Run and Debug
To customize Run and Debug create a launch.json file.
Show all automatic debug configurations.
To learn more about launch.json, see Configuring C/C++ debugging.

C main.c x C ejercicio8.c C ejercicio6.c {} tasks.json C pilas_arreglos.c C pilas_punteros.c C funciones_reutilizables.c
C main.c > main()
5 int main()
23 //IniciarEjercicio7();
24 //IniciarEjercicio8();
25
26 IniciarEjercicio8();
27
28 return 0;

C pilas_punteros.c x
C pilas_punteros.c > TAMANIO_MAXIMO
1 #include "pilas.h"
2 #include <stdlib.h>
3 #include <stdio.h>
4 #include <stdbool.h>
5
6 static const int TAMANIO_MAXIMO = 3;
7
8 struct Nodo {

OUTPUT TERMINAL PORTS
Cargar una pila con valores repetidos para obtener una nueva con los valores y la cantidad de veces que se repiten
Cargar pila:
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 1
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 2
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 3
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 4
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 5
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 7
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): .
Contenido de la pila: 3 2 1
1: 1
2: 1
3: 1
[1] + Done "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm} 0<"/tmp/Microsoft-MIEngine-In-u04pim
pt.ah5" 1>"/tmp/Microsoft-MIEngine-Out-adw3im2q.5oi"
claudiar@claudia:~/programacion2/2024-2/Correcciones manuales/05/programacion2_2024_2c/tp-pilas$
```



```
File Edit Selection View Go Run Terminal Help
RUN AND DEBUG: RUN
Run and Debug
To customize Run and Debug create a launch.json file.
Show all automatic debug configurations.
To learn more about launch.json, see Configuring C/C++ debugging.

C main.c x C ejercicio8.c C ejercicio6.c {} tasks.json C pilas_arreglos.c C pilas_punteros.c C funciones_reutilizables.c
C main.c > main()
5 int main()
23 //IniciarEjercicio7();
24 //IniciarEjercicio8();
25
26 IniciarEjercicio8();
27
28 return 0;

C pilas_punteros.c x
C pilas_punteros.c > TAMANIO_MAXIMO
1 #include "pilas.h"
2 #include <stdlib.h>
3 #include <stdio.h>
4 #include <stdbool.h>
5
6 static const int TAMANIO_MAXIMO = 3;
7
8 struct Nodo {

OUTPUT TERMINAL PORTS
Cargar una pila con valores repetidos para obtener una nueva con los valores y la cantidad de veces que se repiten
Cargar pila:
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 1
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 1
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 2
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 2
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 1
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 2
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): 1
> Ingrese un valor numerico o ( PUNTO ' ' para terminar ): .
Contenido de la pila: 2 1 1
1: 2
2: 1
[1] + Done "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm} 0<"/tmp/Microsoft-MIEngine-In-muwndt
ey.lai" 1>"/tmp/Microsoft-MIEngine-Out-1sin30ka.o0t"
claudiar@claudia:~/programacion2/2024-2/Correcciones manuales/05/programacion2_2024_2c/tp-pilas$
```