

RESULTADO DE LA CORRECCIÓN: **APROBADO-**

## OBSERVACIONES

Incluyen el .c de las implementaciones de listas en .c del ejercicio. Los #include deben ser de .h. No se debería utilizar TAMANIO\_MAXIMO, ya que es una constante propia de la implementación. De hecho, si no hubieran incluido el archivo .c de la implementación, no estaría disponible (misma observación hecha para Listas). No permite pilas vacías. En el ej 6 no me permite la carga manual de datos, la pila con datos al azar que se genera es siempre la misma. Hay que descomentar la función iterativa para poder probarla. En los tres últimos ejercicios usan una función "copiar\_pila" que si bien copia la estructura no genera nuevos "tipo\_elemento". En el ejercicio 6 debieron ejecutar las dos implementaciones o incluir una instancia para elegir cuál se quiere ejecutar.

The screenshot displays the Visual Studio Code interface with the following components:

- File Explorer (Left):** Shows the project structure for 'TP 3 PILAS', including files like 'ejercicio2.c', 'ejercicio3.c', 'ejercicio4.c', 'ejercicio5.c', 'ejercicio6.c', 'ejercicio7.c', 'ejercicio8.c', 'pilas\_arreglos.c', 'pilas\_punteros.c', 'pilas.h', 'tipo\_elemento.c', 'tipo\_elemento.h', 'tp\_pilas.h', and 'TP3.md'.
- Main Editor:** Displays the code for 'ejercicio6.c'. The visible code includes a function 'p\_ej6\_eliminarclave(Pila p, int clave)' and a function 'Pila copiar\_pila(Pila pilaOriginal)'. The 'copiar\_pila' function creates a new stack and copies elements from the original stack.
- Terminal (Bottom):** Shows the execution of the program. It displays the initial stack content: '[PILA ORIGINAL] Contenido de la pila: 1 4 2 1 0 3 0 2 1 3'. It then shows the user input '0' to remove an element, and the updated stack content: '[ELIMINAR] Ingrese un numero entero: 0 Eliminando elemento/s con clave 0 Contenido de la pila: 1 4 2 1 3 2 1 3'. The terminal also shows the program's exit status: 'The program "/home/clauidar/programacion2/2024/correccionesManuales/grupo8/TP 3 PILAS/ejercicio6" has exited with code 122 (0x000000c)'.

## GRUPO 8 – Correcciones Trabajo Práctico: PILAS

The image shows a Visual Studio Code editor with a C program for stack operations. The program is named `ejercicio6.c` and is located in the `TP3 PILAS` directory. The code implements a stack with functions for creating, copying, and deleting elements. The terminal output shows the program's execution, including the initial stack state, the deletion of an element, and the final stack state.

```
File Edit Selection View Go Run Terminal Help
EXPLORER
TP3 PILAS
.vscode
tasks.json
COMPLEJIDAD.md
ejercicio2.c
ejercicio3.c
ejercicio4.c
ejercicio5.c
ejercicio6.c
ejercicio7.c
ejercicio8.c
pilas_arreglos.c
pilas_punteros.c
pilas.h
tipo_elemento.c
tipo_elemento.h
TP3.md

ejercicio6.c
// funcion que copia una pila sin alterar la original
Pila copiar_pila(Pila pilaOriginal) {
    Pila pilaAuxiliar = p_crear();
    Pila pilaCopia = p_crear();
    while (!p_es_vacia(pilaOriginal)) {
        TipoElemento elemento = p_desapilar(pilaOriginal);
        p_apilar(pilaAuxiliar, elemento);
    }
    while (!p_es_vacia(pilaAuxiliar)) {
        TipoElemento elemento = p_desapilar(pilaAuxiliar);
        p_apilar(pilaOriginal, elemento);
        p_apilar(pilaCopia, elemento);
    }
}

TERMINAL
[PILA ORIGINAL]
Contenido de la pila: 1 4 2 1 0 3 0 2 1 3

[ELIMINAR]
Ingrese un numero entero: 1
Eliminando elemento/s con clave 1
Contenido de la pila: 4 2 0 3 0 2 3

[PILA ORIGINAL]
Contenido de la pila: 1 4 2 1 0 3 0 2 1 3

[1] + Done
"/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm} 0<
"/tmp/Microsoft-MIEngine-In-0ckdvbrl.nnb" 1>"/tmp/Microsoft-MIEngine-Out-fzwzvgcs.dd
n"
claudiar@claudia:~/programacion2/2024/correccionesManuales/grupo8/TP 3 PILAS$
```