

Tipos de Datos Abstractos (TDA) - COLA.

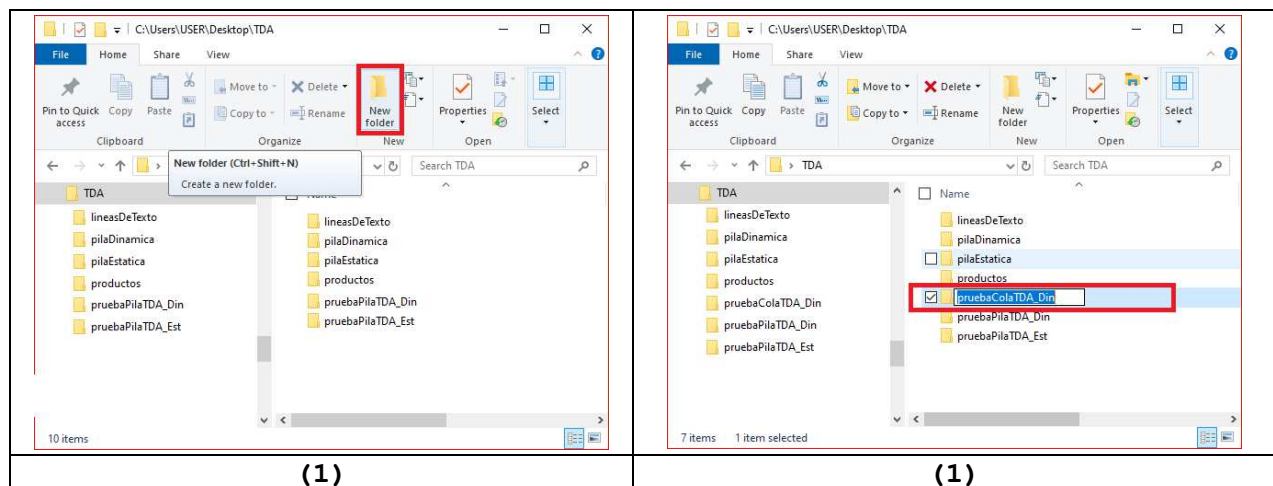
Agregar los proyectos de prueba para memoria estática y dinámica.

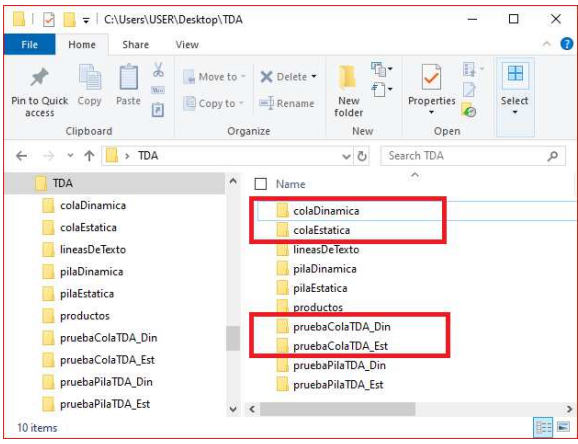
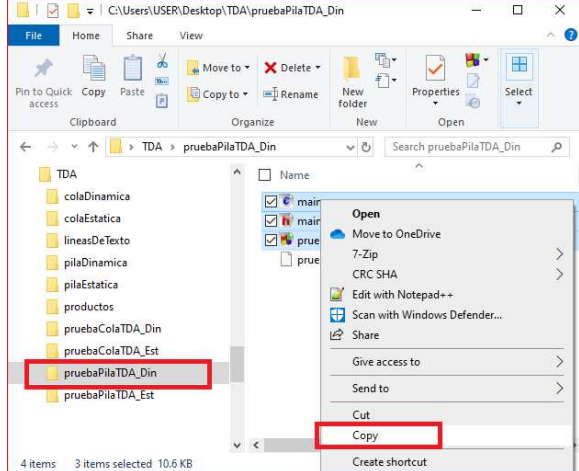
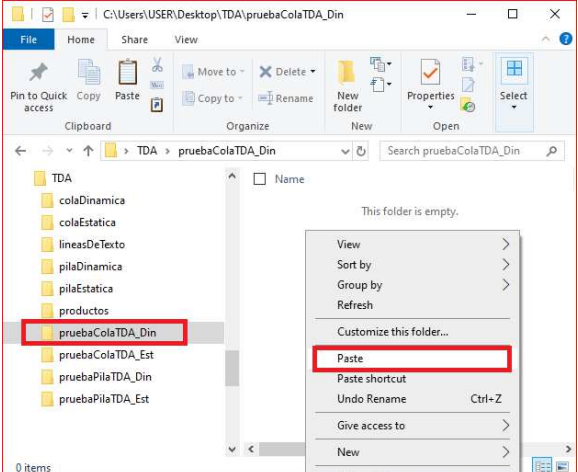
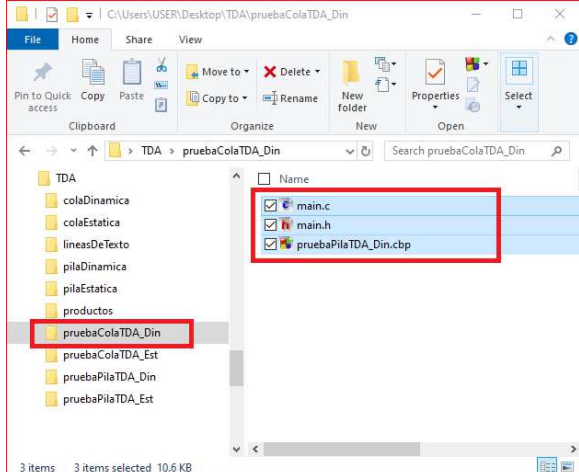
En la carpeta TDA proceda a generar los proyectos "pruebaColaTDA_Din" y "pruebaColaTDA_Est".

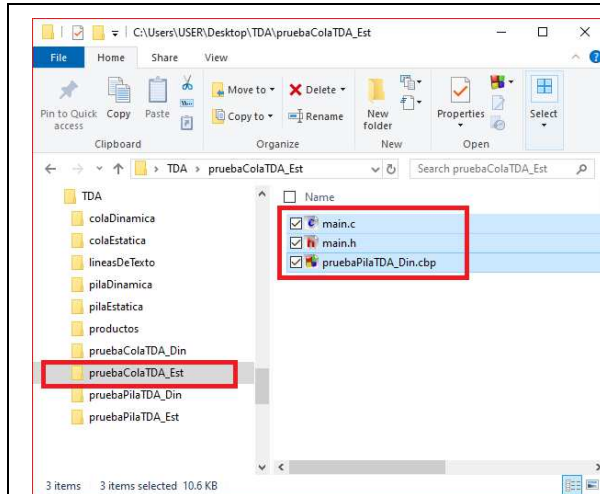
Puede hacerlo con el IDE de Code::Blocks o recuerde que, puede:

- generar las carpetas "pruebaColaTDA_Din" y "pruebaColaTDA_Est" además de "colaDinamica" y "colaEstatica" con el explorador de Windows (1),
- copiar y renombrar el archivo ".cbp" de uno de los proyectos anteriores, por ejemplo: "pruebaPilaTDA_Din" (2) y copiar además los archivos "main.c" y "main.h",
- modificar el nombre de archivo para que sea "pruebaColaTDA_Din" (3)
- editarlo y modificarle en el código "xml" todas las ocurrencias de, por ejemplo, "pruebaPilaTDA_Din" por "pruebaColaTDA_Din" (4).

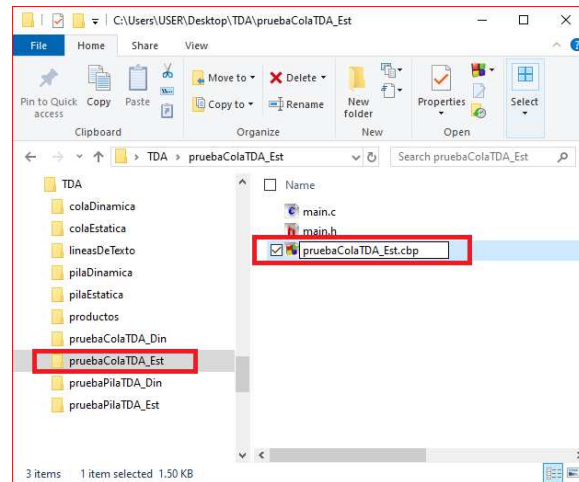
Luego hacer lo mismo para el otro proyecto.



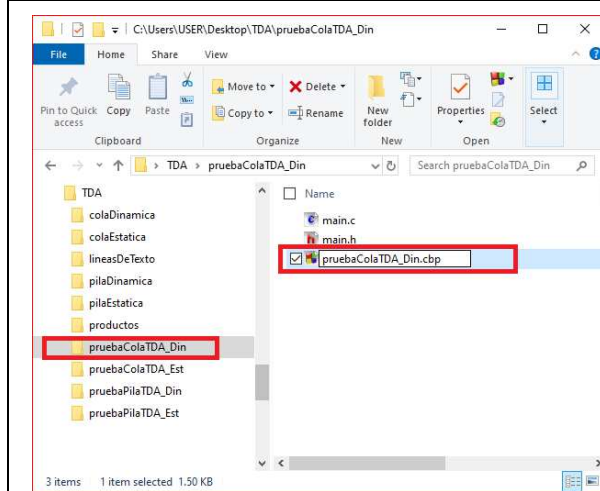
	
(1)	(2)
	
(2)	(2)



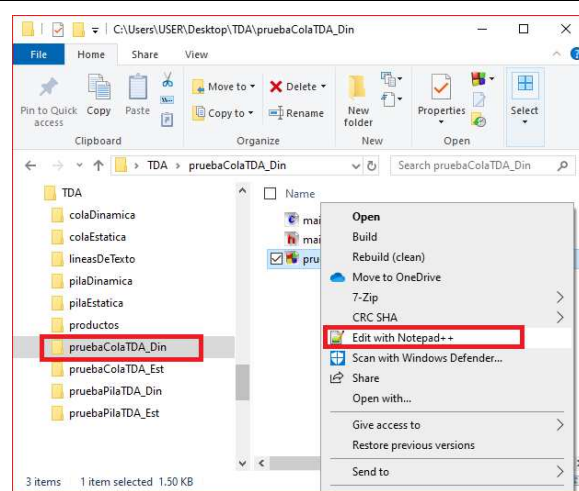
(2)



(3)



(3)



(4)



UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

The screenshot shows a Notepad++ window editing an XML file named 'pruebaColaTDA_Din.cbp'. The XML code is as follows:

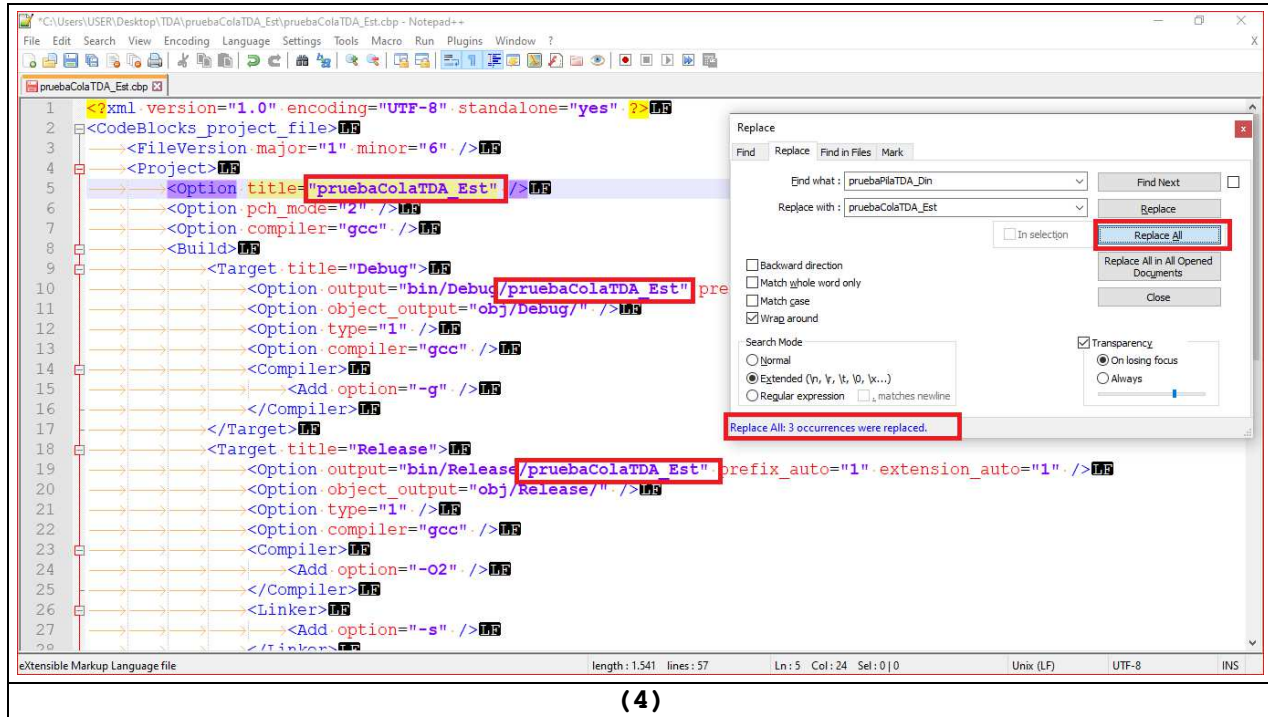
```

1 <?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
2 <CodeBlocks_project_file>
3   <FileVersion major="1" minor="6" />
4   <Project>
5     <Option title="pruebaPilaTDA_Din" />
6     <Option pch_mode="2" />
7     <Option compiler="gcc" />
8     <Build>
9       <Target title="Debug">
10        <Option output="bin/Debug/pruebaPilaTDA_Din" prefix_auto="1" extension_auto="1" />
11        <Option object_output="obj/Debug/" />
12        <Option type="1" />
13        <Option compiler="gcc" />
14        <Compiler>
15          <Add option="-g" />
16        </Compiler>
17      </Target>
18      <Target title="Release">
19        <Option output="bin/Release/pruebaPilaTDA_Din" prefix_auto="1" extension_auto="1" />
20        <Option object_output="obj/Release/" />
21        <Option type="1" />
22        <Option compiler="gcc" />
23        <Compiler>
24          <Add option="-O2" />
25        </Compiler>
26        <Linker>
27          <Add option="-s" />
28        </Linker>
29      </Target>
30    </Build>
31  </Project>
32</CodeBlocks_project_file>

```

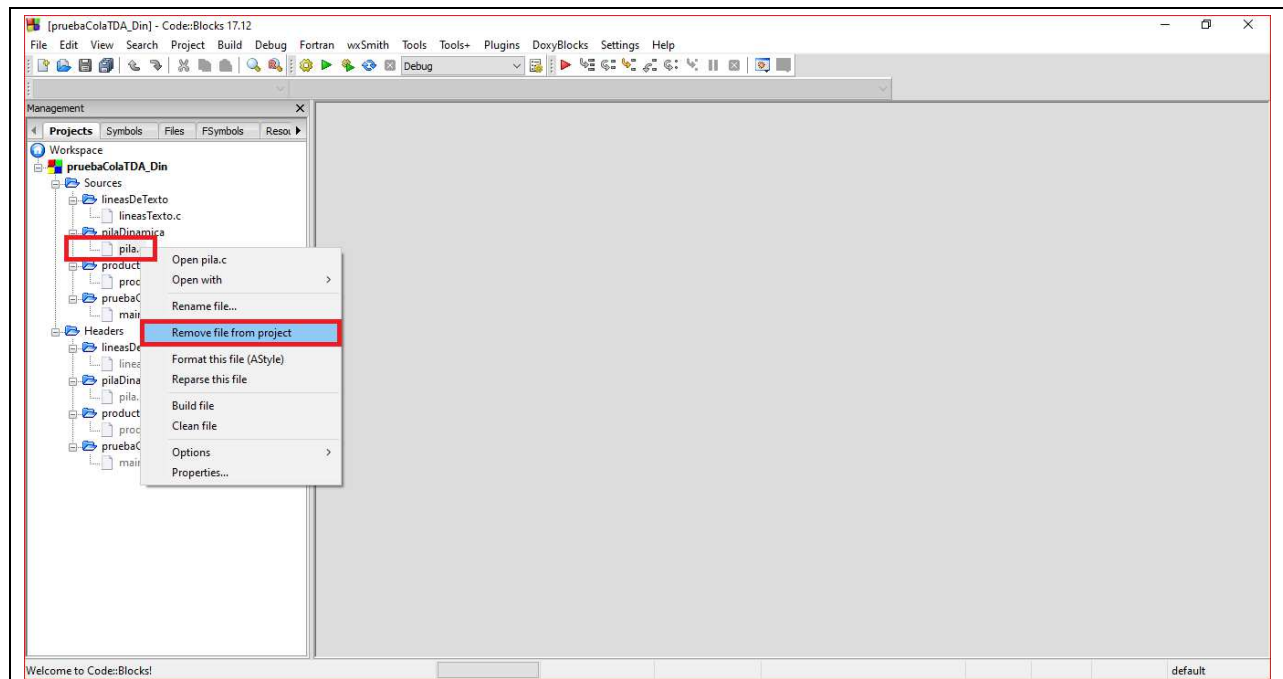
A 'Replace' dialog box is open, showing the replacement of 'pruebaPilaTDA_Din' with 'pruebaColaTDA_Din'. The 'Replace All' button is highlighted. The status bar at the bottom indicates the file is an eXtensible Markup Language file, 1,541 characters long, 57 lines, and uses UTF-8 encoding.

(4)

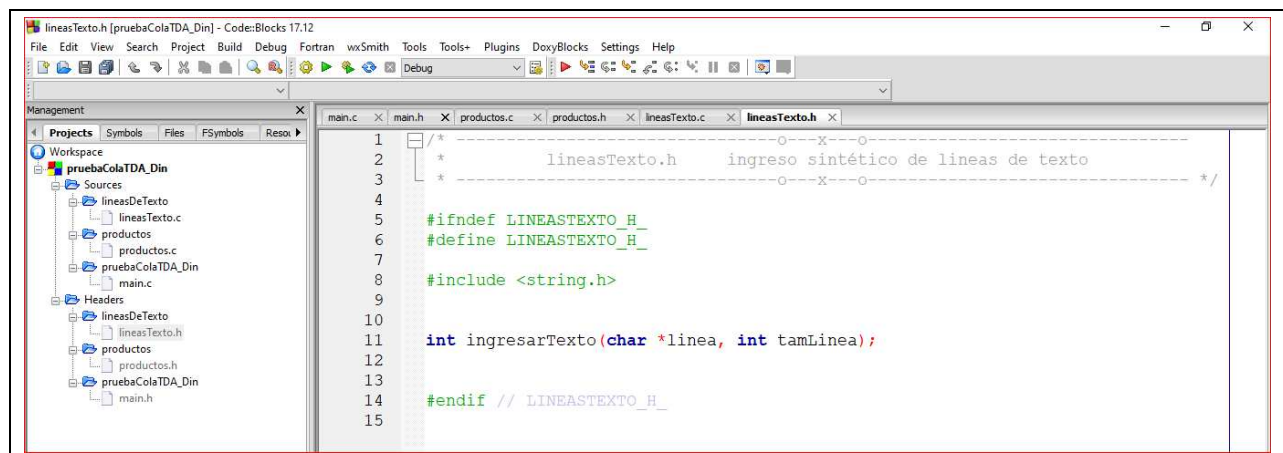


(4)

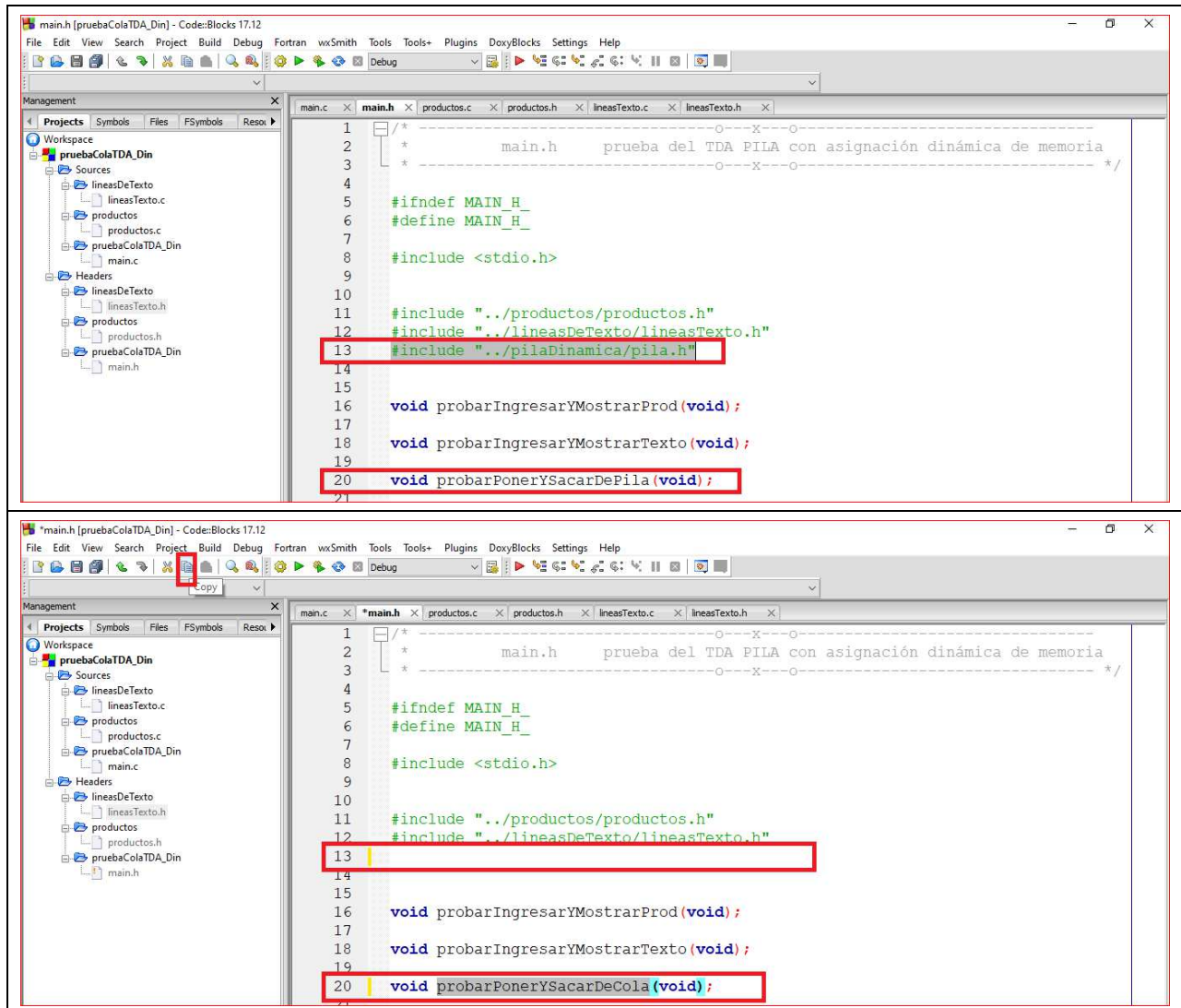
Abra ambos proyectos y remueva de los mismos los fuentes correspondientes a "pila.c" y "pila.h".



Abra (haciendo doble-click en el panel izquierdo), los archivos fuente "main.c", "main.h", "productos.c", "productos.h", "lineasTexto.c" y "lineasTexto.h"



Edite en el IDE el archivo fuente que corresponde a "main.h" y elimine del mismo la directiva "#include "../pilaDinamica/pila.h", y modifique el prototipo de la función de prueba, para que sea consistente ("probarPonerYSacarDeCola"). Copie el nombre de la función . . .



```
1  /*-----X-----*/
2  *      main.h      prueba del TDA PILA con asignación dinámica de memoria
3  *      *-----X-----*/
4
5  #ifndef MAIN_H_
6  #define MAIN_H_
7
8  #include <stdio.h>
9
10
11 #include "../productos/productos.h"
12 #include "../lineasDeTexto/lineasTexto.h"
13 #include "../pilaDinamica/pila.h"
14
15 void probarIngresarYMostrarProd(void);
16
17 void probarIngresarYMostrarTexto(void);
18
19 void probarPonerYSacarDePila(void);
20
21
```

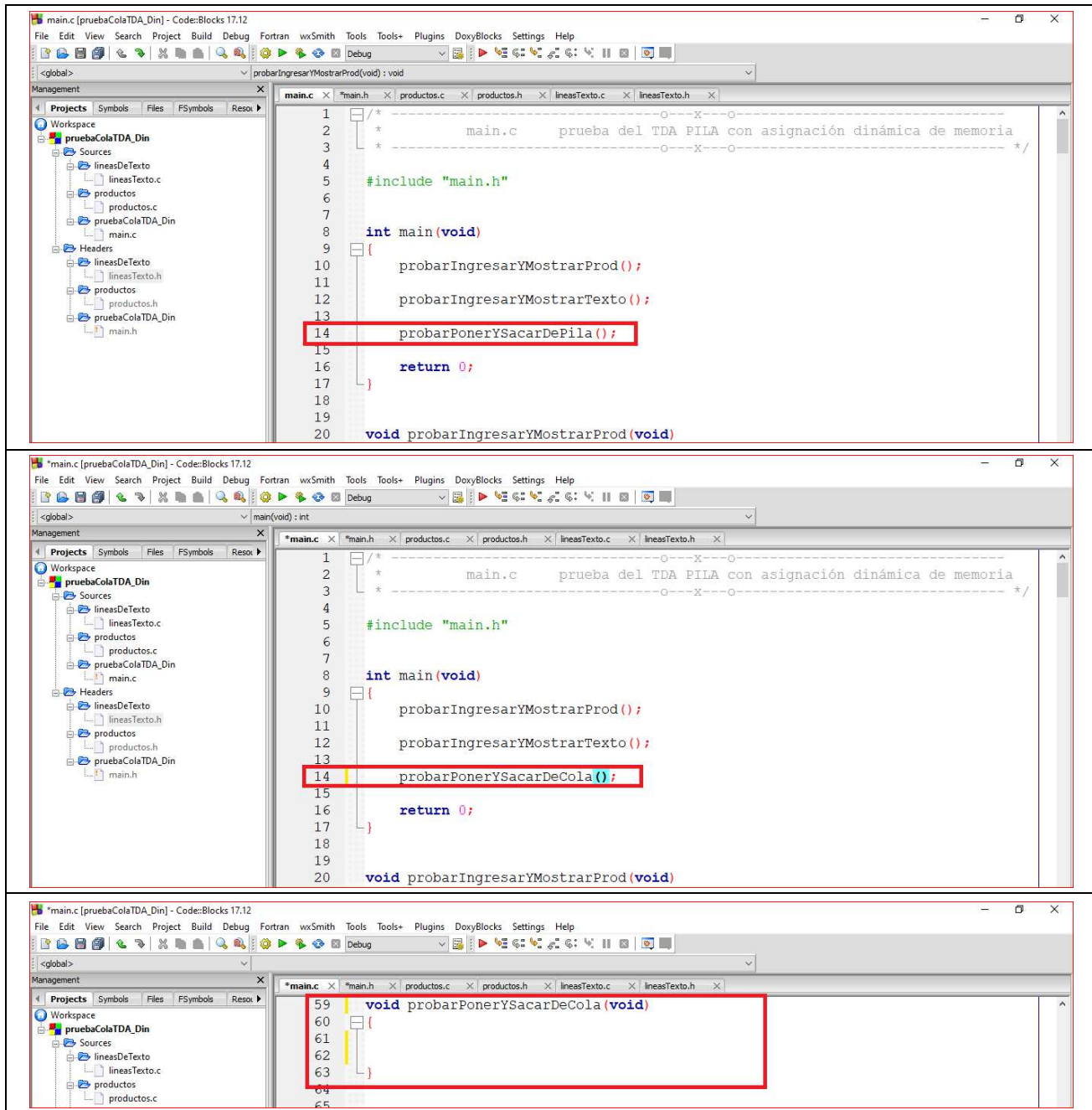
```
1  /*-----X-----*/
2  *      main.h      prueba del TDA PILA con asignación dinámica de memoria
3  *      *-----X-----*/
4
5  #ifndef MAIN_H_
6  #define MAIN_H_
7
8  #include <stdio.h>
9
10
11 #include "../productos/productos.h"
12 #include "../lineasDeTexto/lineasTexto.h"
13
14
15 void probarIngresarYMostrarProd(void);
16
17 void probarIngresarYMostrarTexto(void);
18
19 void probarPonerYSacarDeCola(void);
20
21
```



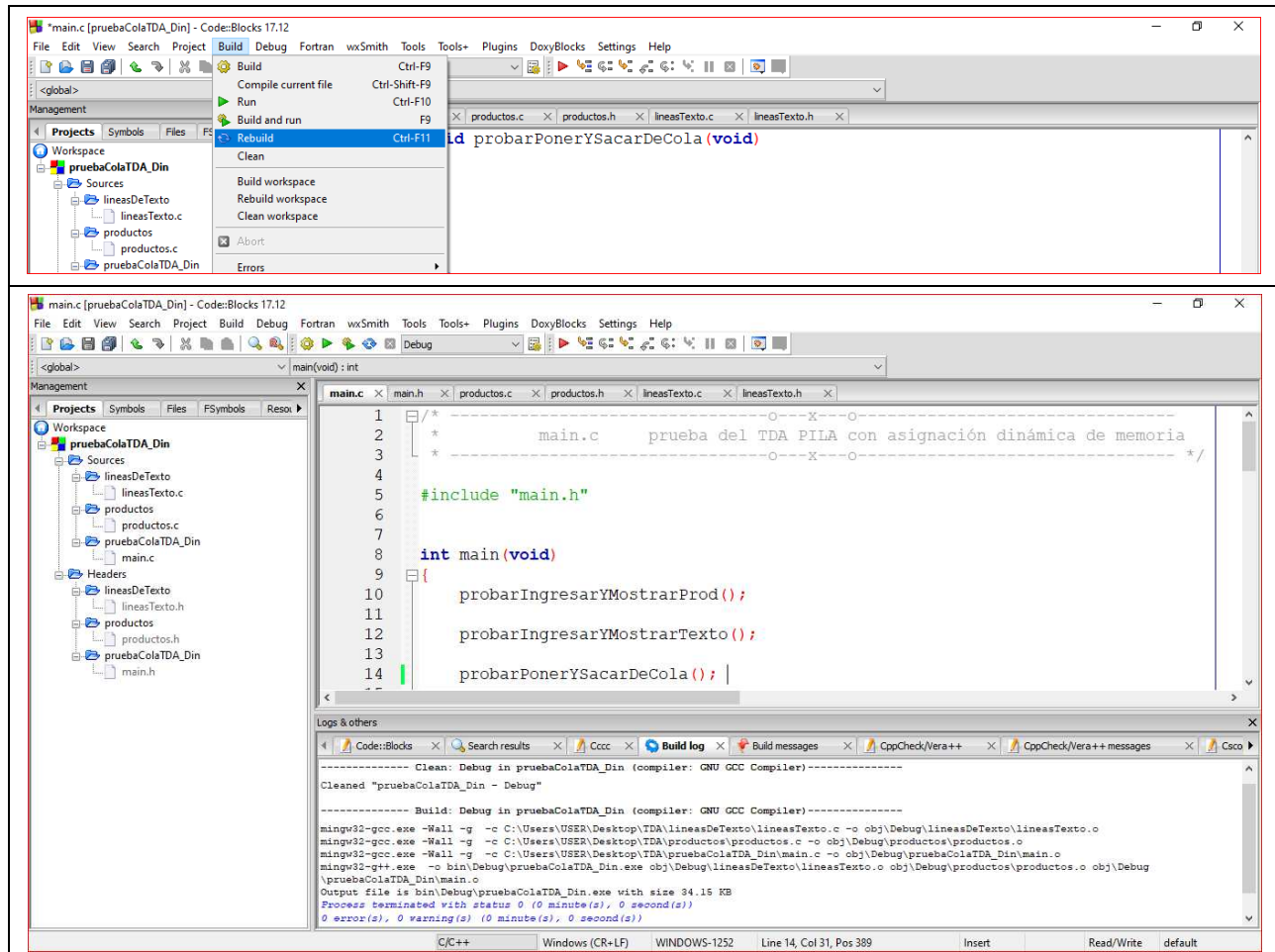
UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

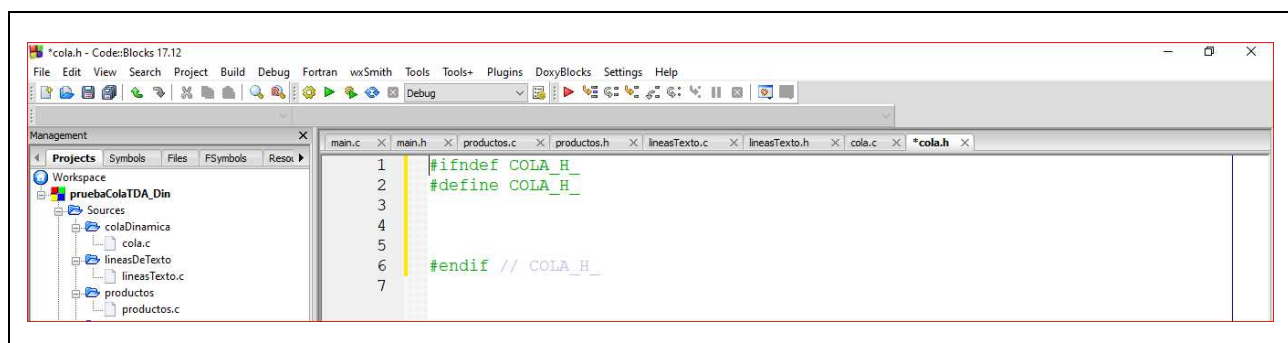
. . . para, editando el archivo "main.c", reemplazar la invocación y el desarrollo de la función de prueba. Además, en el desarrollo de la función de prueba elimine el código de la misma . . .



Compile y asegúrese que lo hecho hasta ahora no produce ningún problema.



Agregue al proyecto los archivos fuente correspondientes a "cola.c" y "cola.h" en la carpeta "colaDinamica" creada al comienzo (también las podría crear con el IDE).

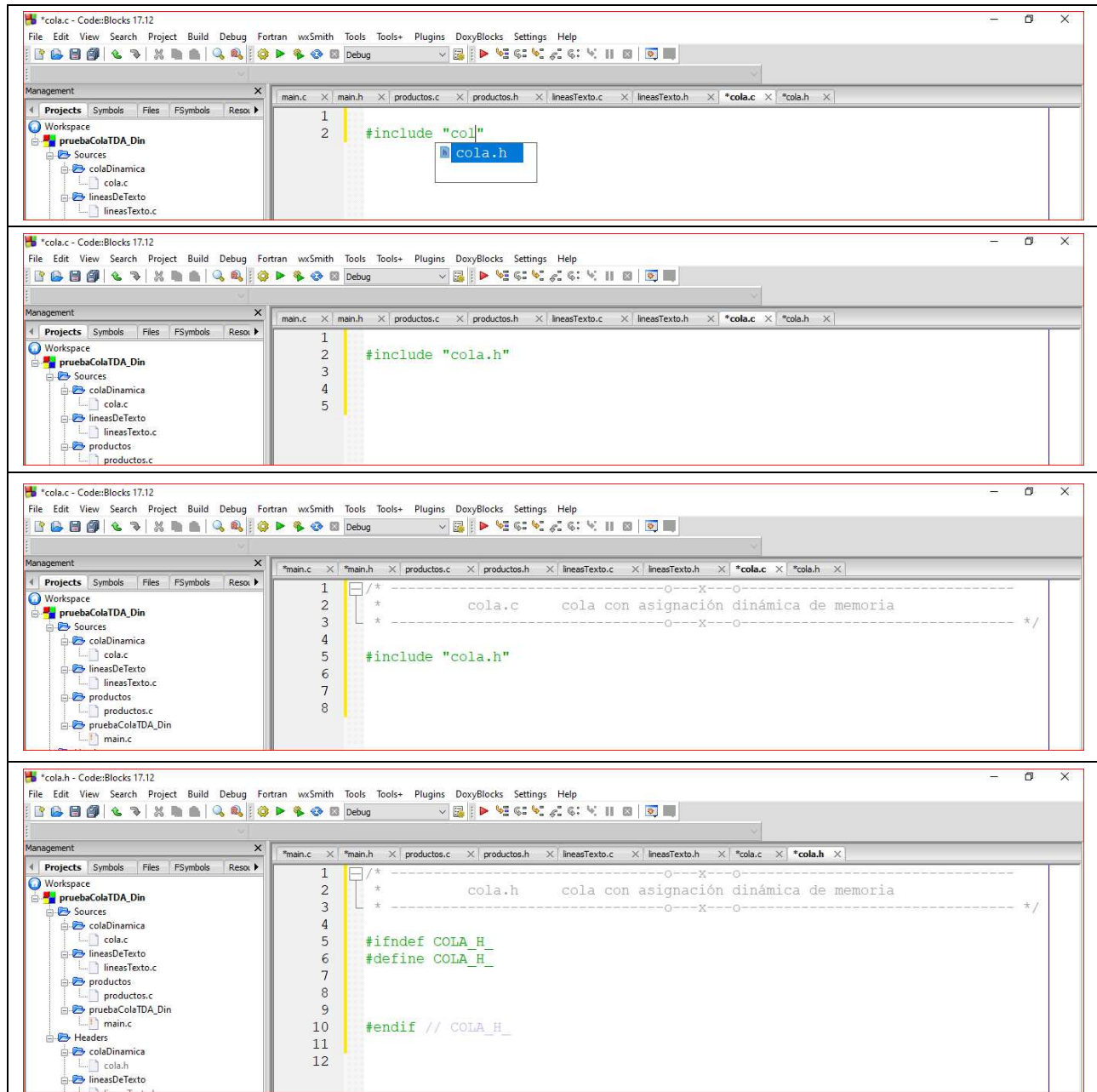




UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

Vaya a cola.c con el IDE y agréguele a ambos fuentes . . .



Compile asegurándose que no hay "warnings" ni errores, ejecute y vea que, por ahora, este proyecto hace lo que debe . . .



UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

Proceda del mismo modo con el otro proyecto ("**pruebaColaTDA_Est**"), agregando los archivos fuente de cola ("**cola.c**" y "**cola.h**"), en la carpeta "**colaEstatica**".

Pase a ver el siguiente documento con la explicación de la implementación estática y dinámica de cola para luego codificar las correspondientes primitivas y codificarlas.