|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **APUNTADORES** | | | **No.** | **1** |
| **Asignatura:** | **METODOS NUMERICOS** | **Carrera:** | **ING. SISTEMAS** | **Duración de la práctica (Horas)** | **3** |

**ANDREA YOCELINE ALVAREZ SUAREZ 3402**

1. **Competencia(s) específica(s):**
2. **Desarrollo de la Practica:**
3. **Material utilizado:**

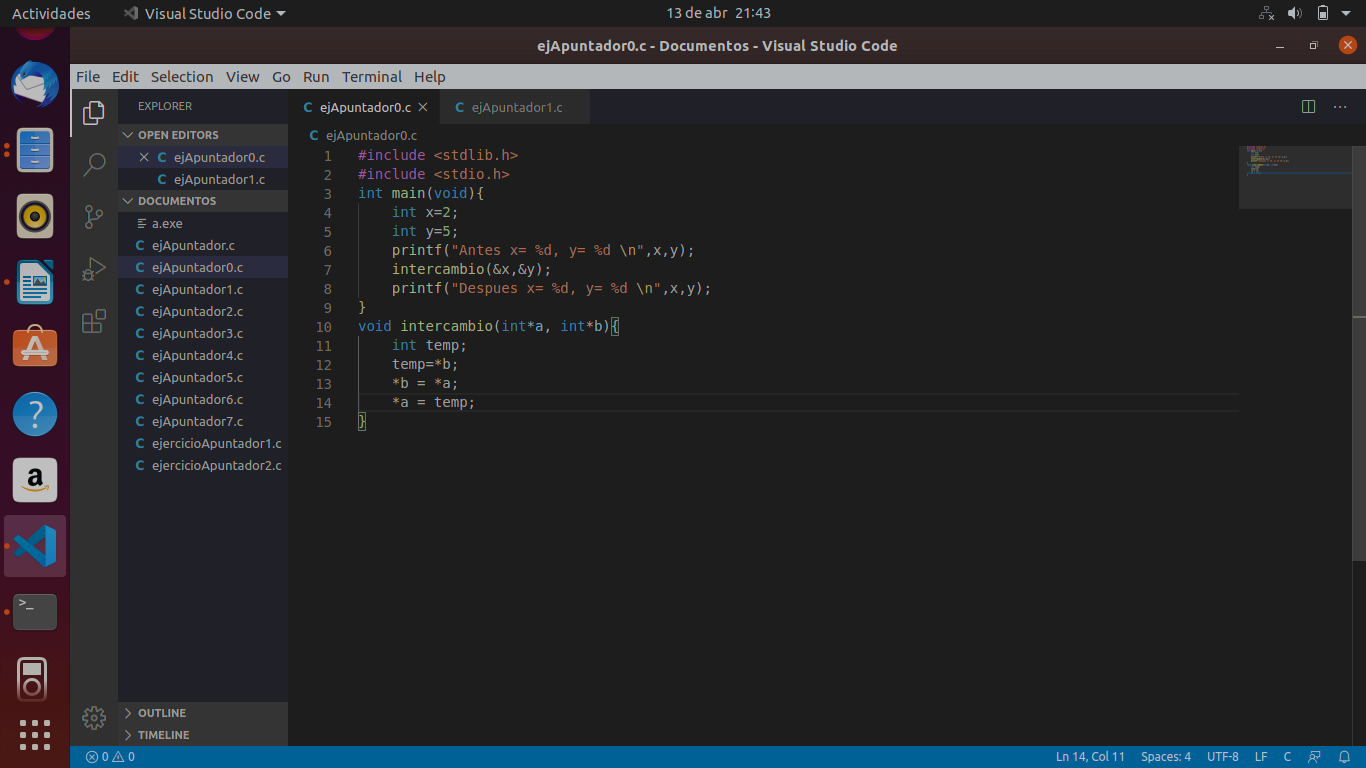
* Visual Studio Code

**¿Qué es un Apuntador?**

Un puntero es un objeto que apunta a otro objeto. Es decir, una variable cuyo valores la dirección de memoria de otra variable.

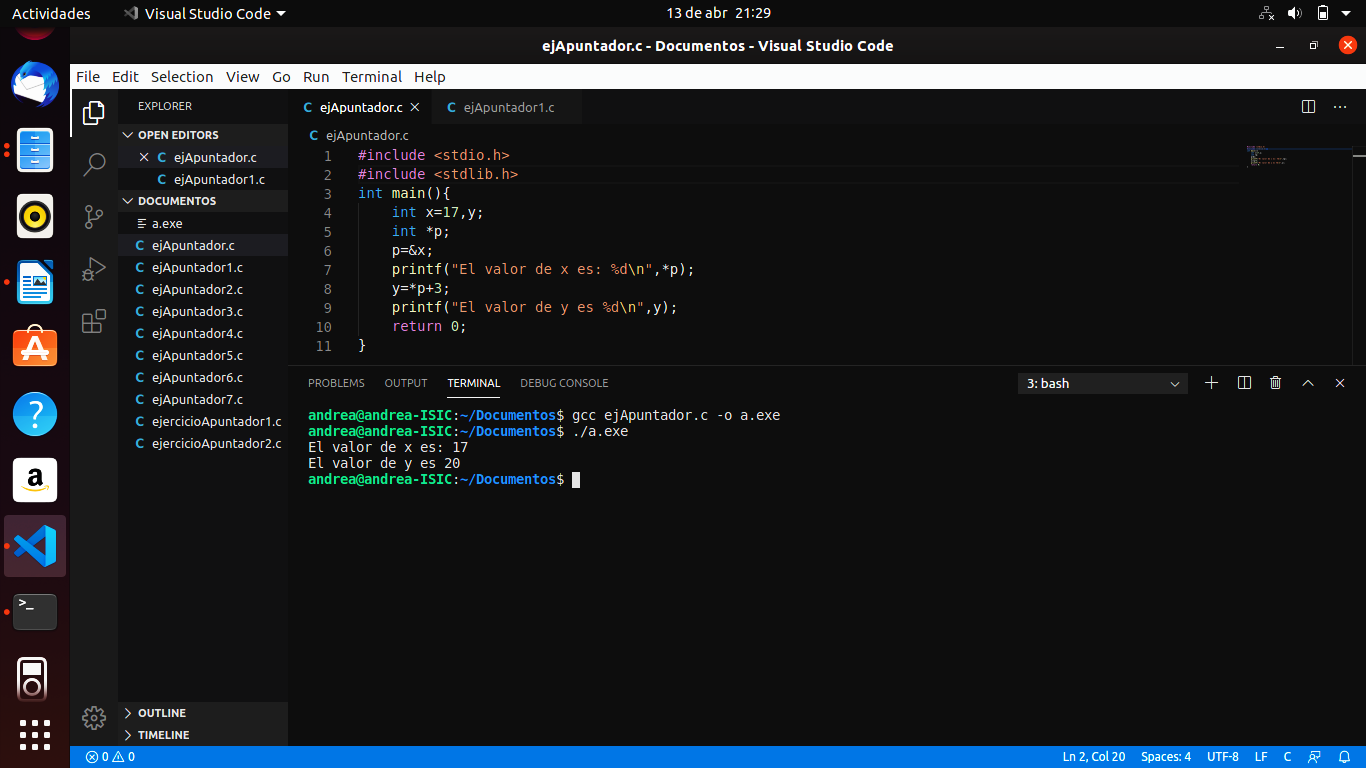
**Ejemplo 1:**

Paso de parámetros por referencia: En este tipo de llamadas los argumentos contienen direcciones de variables. Dentro de la función la dirección se utiliza para acceder al argumento real. En las llamadas por referencia cualquier cambio en la función tiene efecto sobre la variable cuya dirección se pasó como argumento.



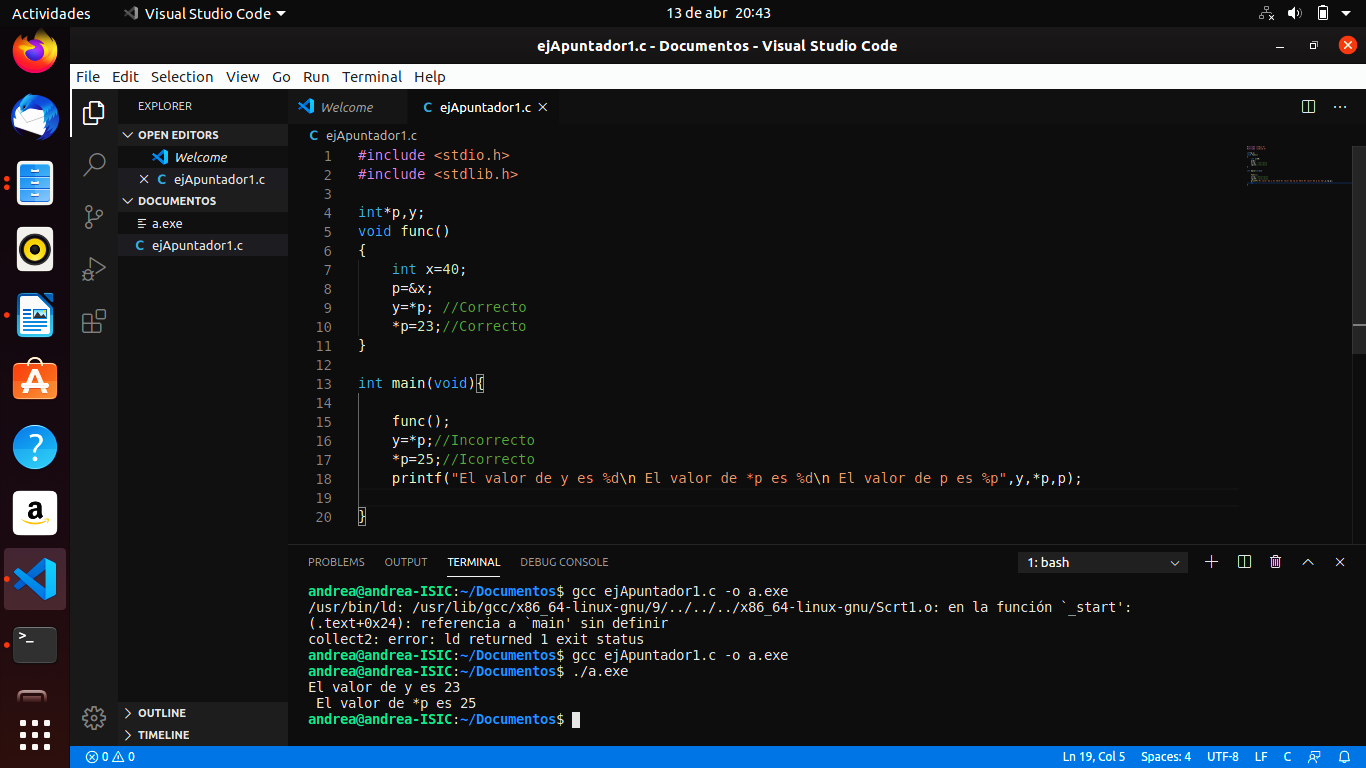
**Ejemplo 2:**

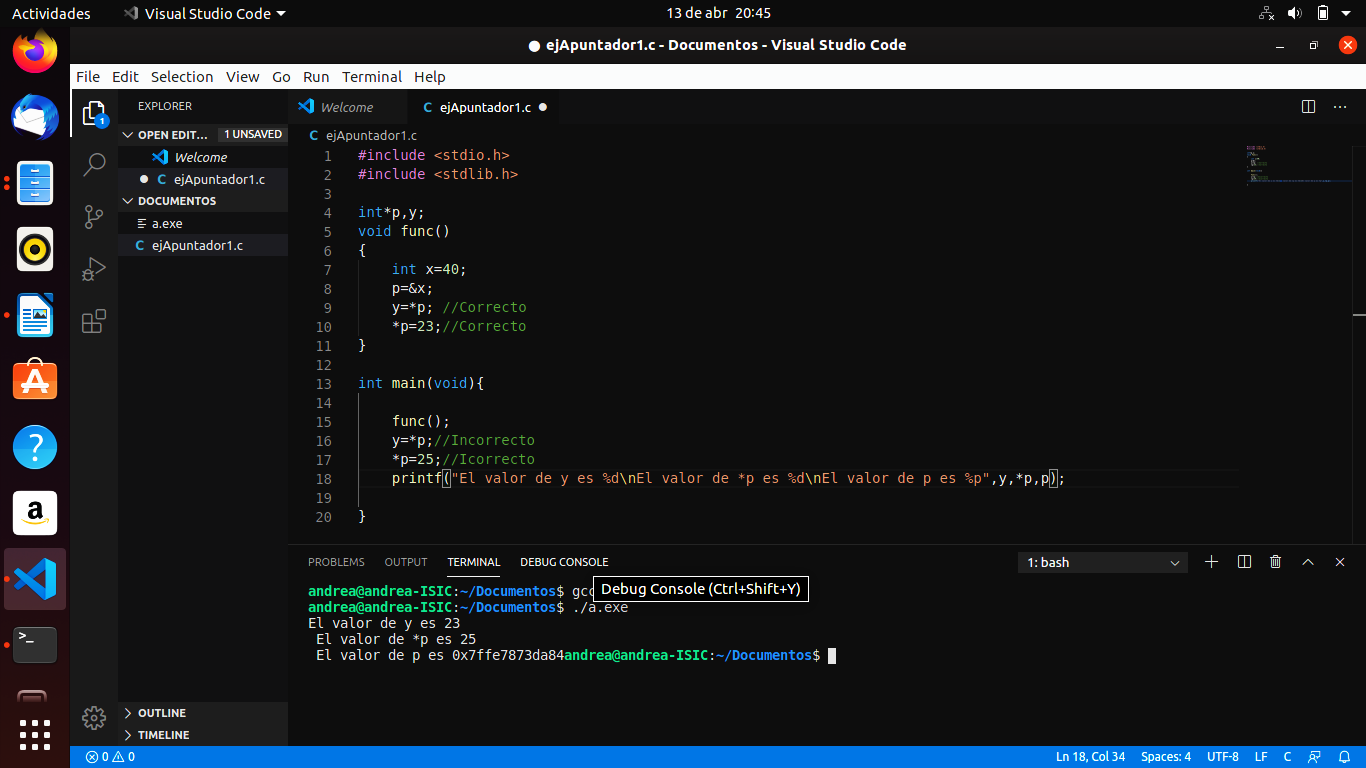
Desreferenciación: Es la obtención del valor almacenado en el espacio de memoria donde apunta un apuntador **(\*p).**



**Ejemplo 3**:

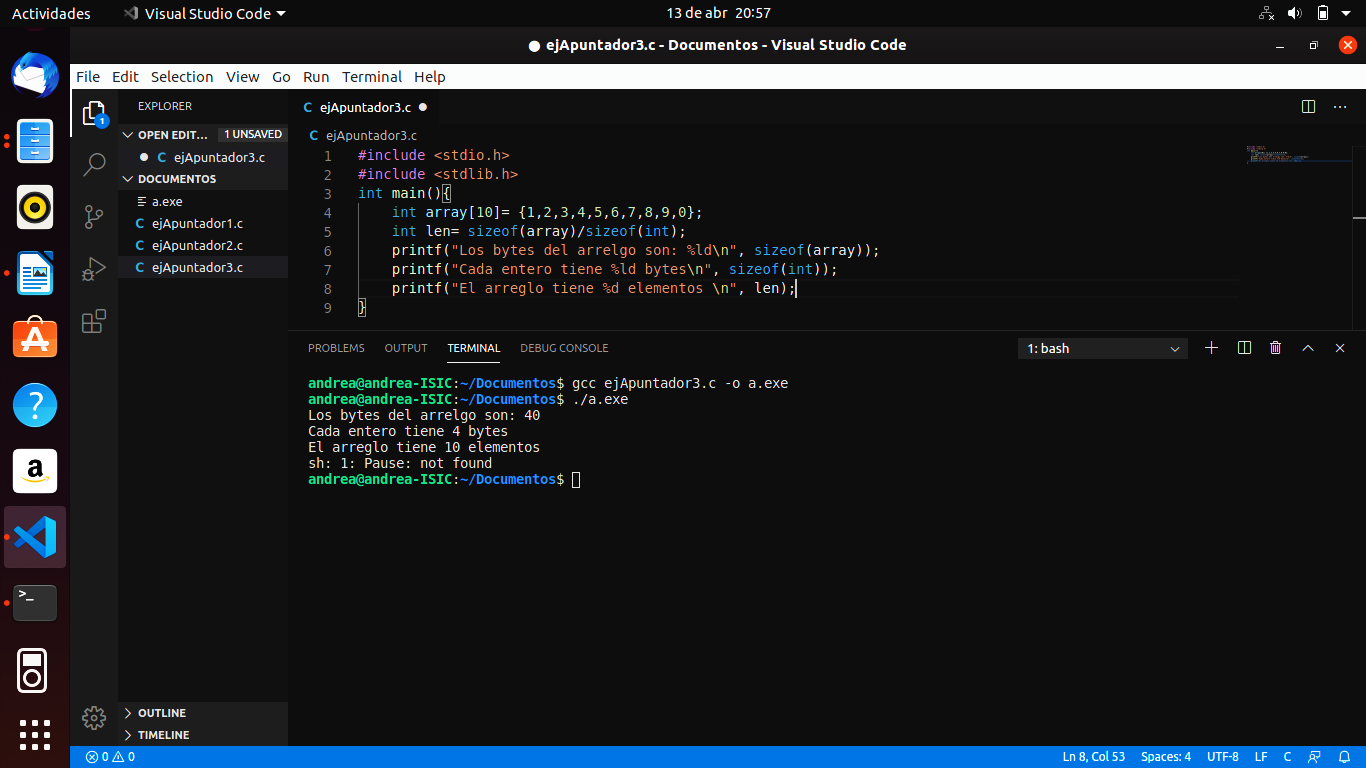
A un apuntador se pueden asignar direcciones de Variables a través del operador de referenciación (‘&’) o direcciones almacenadas en otros apuntadores.





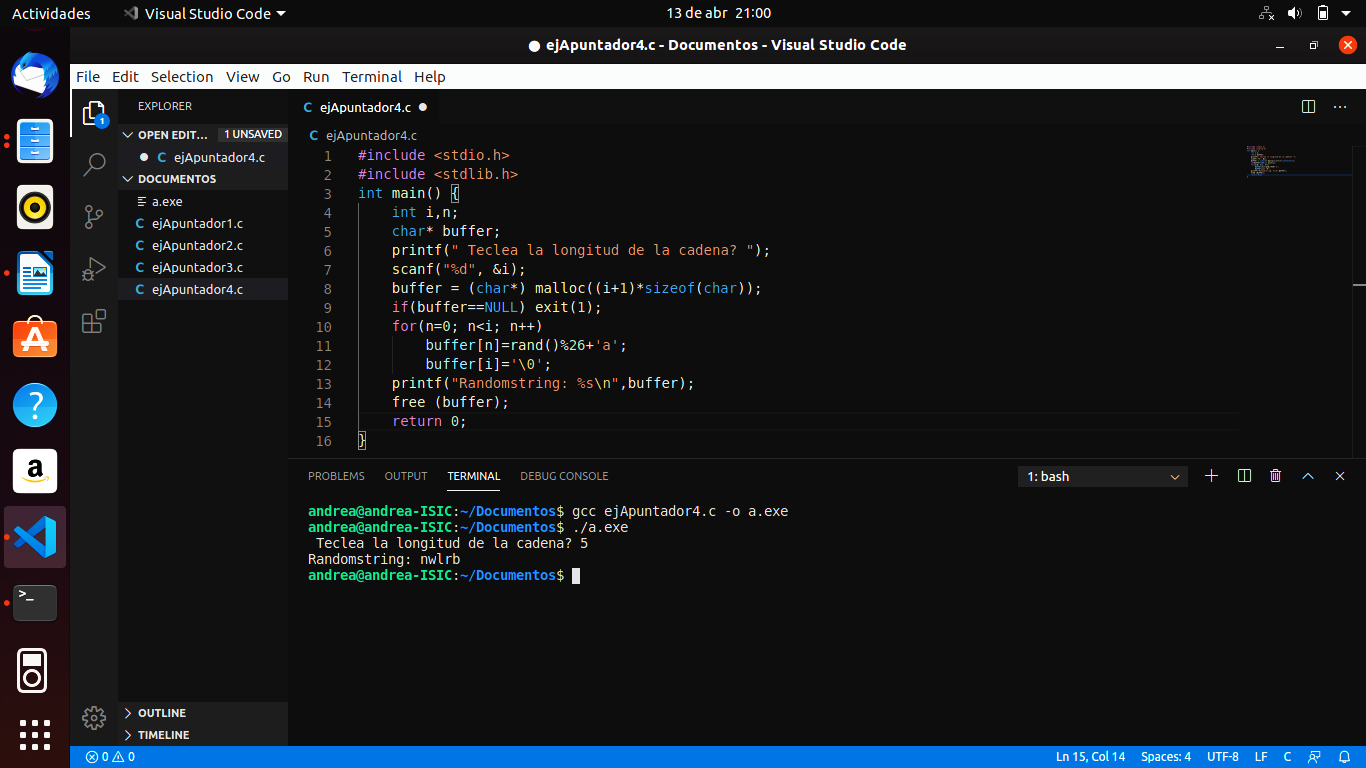
**Ejemplo 4:**

Muestra la cantidad de bytes que contiene el arreglo que está siendo declarado, la cantidad de bytes que tiene cada entero y la cantidad de elementos del arreglo.



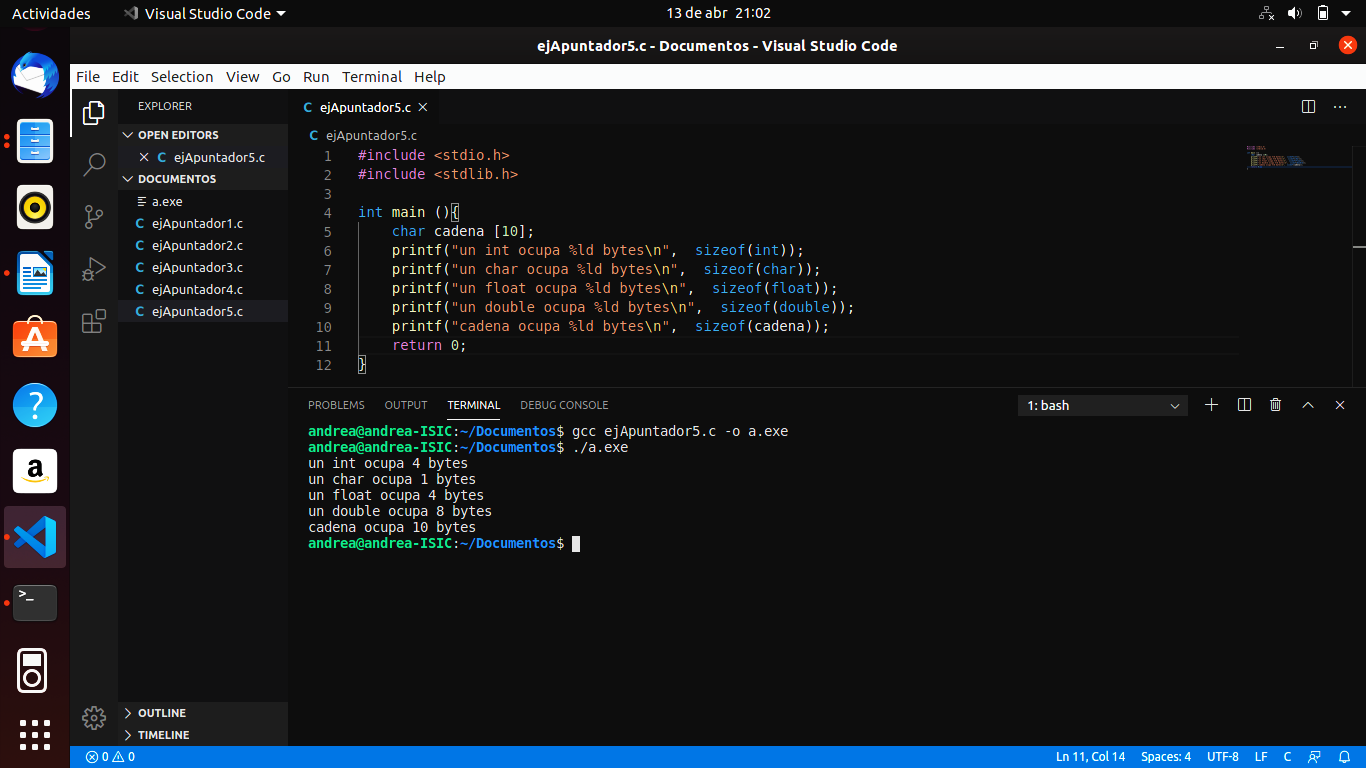
**Ejemplo 5:**

NULL es una macro típicamente definida en archivos de cabecera como stdef.hy stdlib.h.

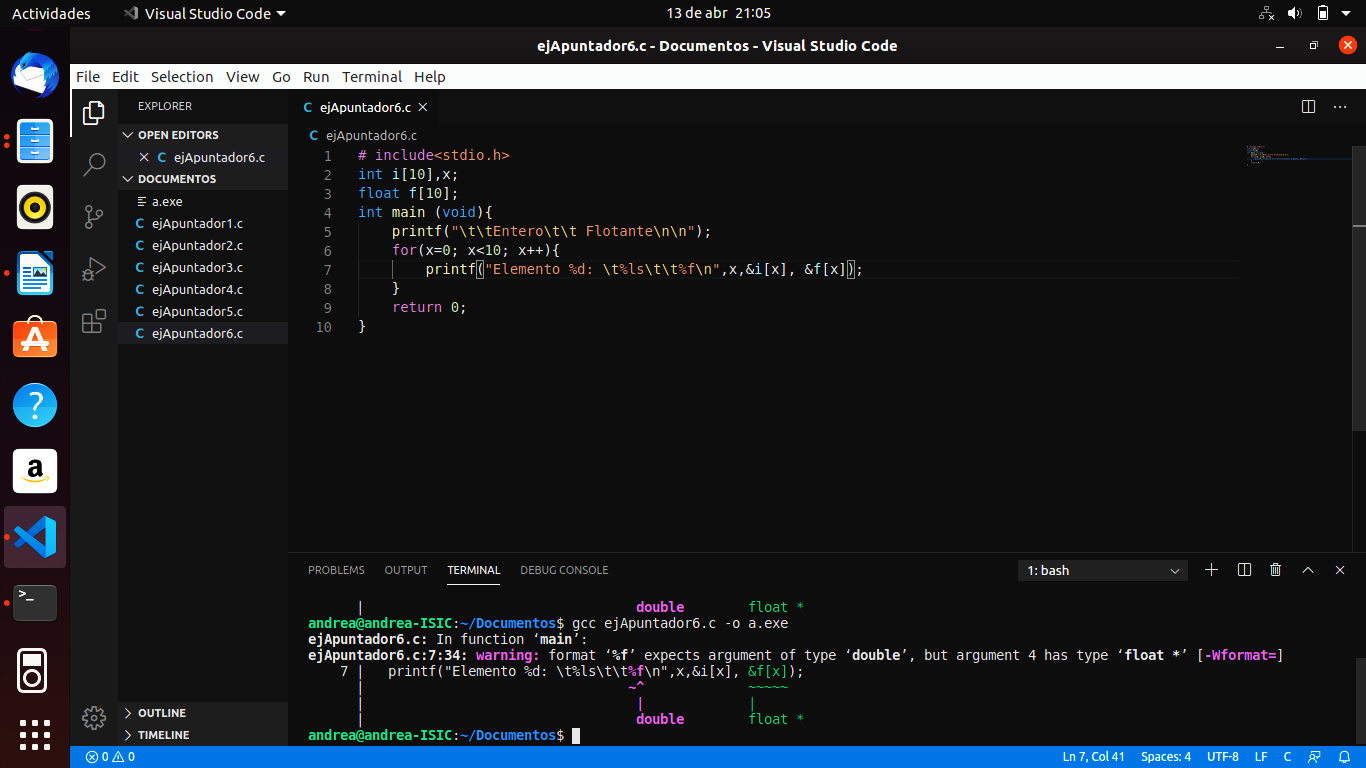


**Ejemplo 6:**

Devuelve el tamaño en bytes que ocupa un tipo o variable en memoria.

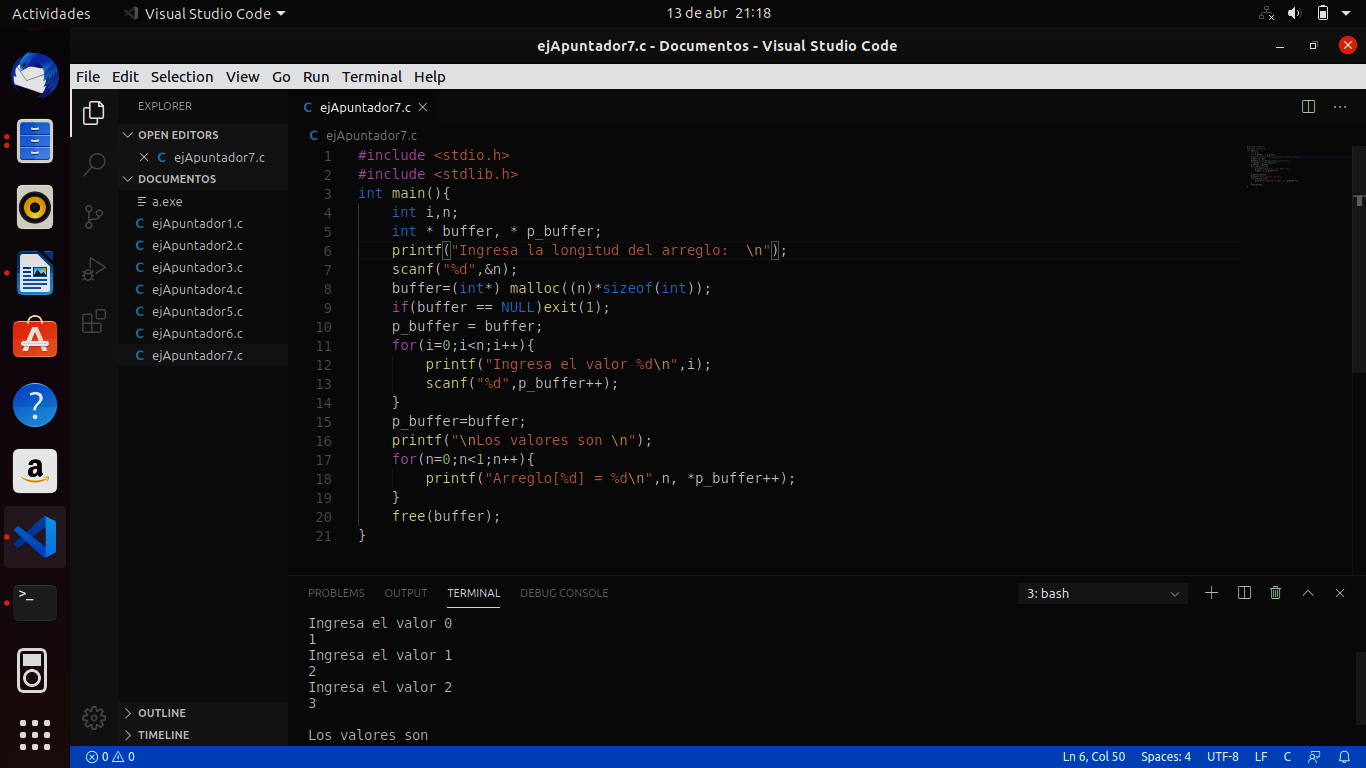


**Ejemplo 7:**



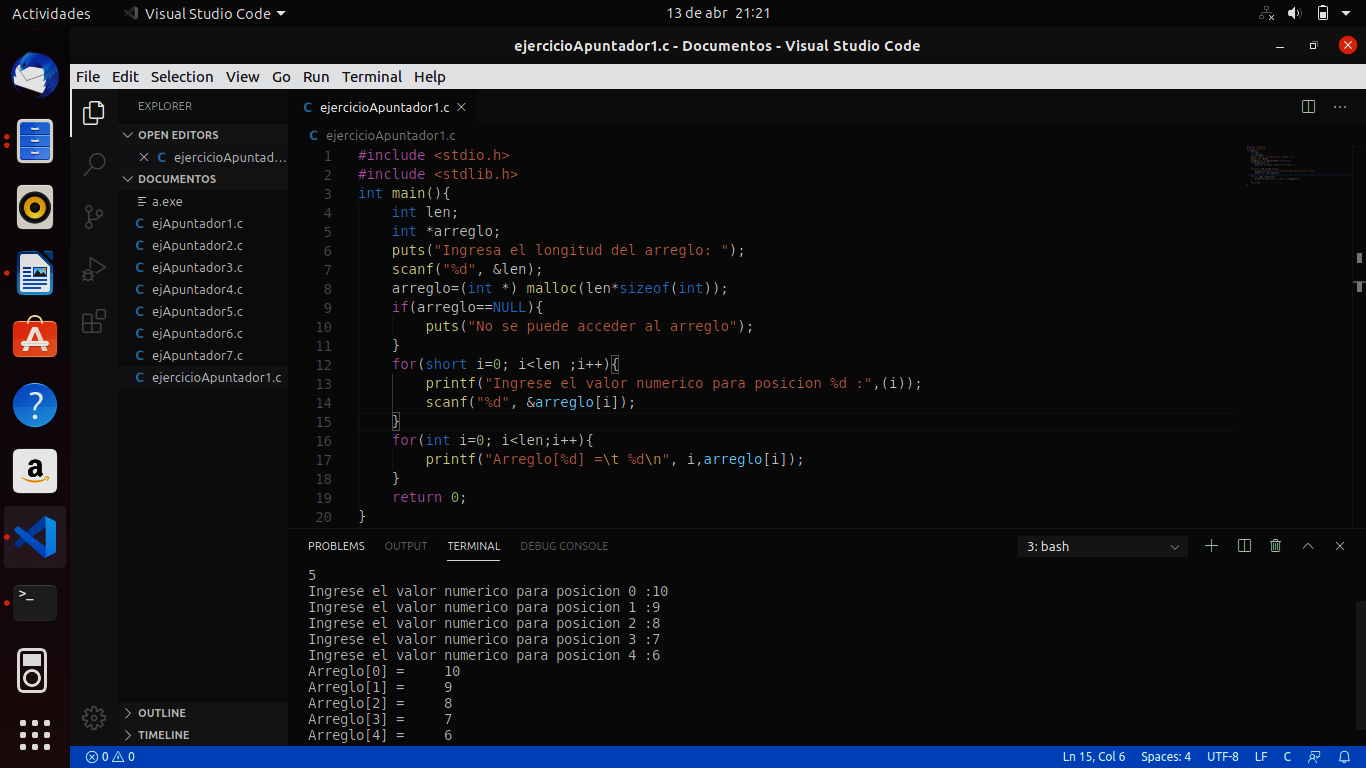
**Ejemplo 8:**

La función malloc() reserva memoria y retorna su dirección, o retorna NULL en caso de no haber conseguido suficiente memoria.



**Ejercicio 1:**

Crea un arreglo de tipo char de tamaño x, en donde x es ingresado por teclado.



**Ejercicio 2:**

Llena elemento por elemento del arreglo con letras ingresados por el usuario. Muestra el arreglo impreso en forma inversa.

