**1. Identificar a qué tipo de direccionamiento pertenece el código que se presenta a continuación.**

**2. Realizar un código similar al código de prueba, sin embargo utiliza cada uno de los métodos de direccionamiento vistos en clase (registro, inmediato, directo, registro indirecto, base mas índice, relativo a registro, relativo a base mas indice)**

**3. Cada código debe de tener sus bloques .date \_start and .text.**

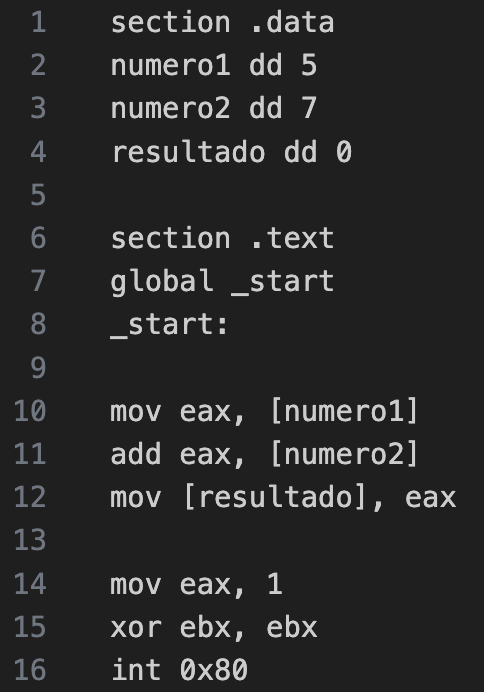
**4. Cada línea de código debe tener un comentario adicional que explique su funcionamiento.**

**5. Al final de cada código, explicar el funcionamiento del modo de direccionamiento utilizado.**

**6. Cada uno de los códigos debe ejecutarse sin errores.**

**7. Realizar una conclusión de al menos 4 párrafos que explique lo aprendido en el taller (modos de direccionamiento, ventas y desventajas de cada uno, por que usar uno en lugar de otro, etc.).**

**8. Documentar su desarrollo, resultados y conclusión en este documento.**

**9. Incluir referencias (si aplica).  
  
**

**1.-** Directo

****

**Explicación del modo de direccionamiento utilizado:**

**Registro (registro-directo):** Se accede directamente a los valores almacenados en los registros de la CPU.

**Inmediato (inmediato):** Los valores están incrustados directamente en las instrucciones.

**Directo (directo):** Se accede directamente a los valores almacenados en una dirección de memoria específica.

**Registro indirecto (registro-indirecto):** Se accede a los valores a través de la dirección almacenada en un registro.

**Base más índice (base-mas-indice):** Se calcula la dirección sumando una base y un índice.

**Relativo a registro (registro-relativo):** Se accede a los valores utilizando una dirección relativa a la dirección almacenada en un registro.

**Relativo a base más índice (base-mas-indice-relativo):** Se accede a los valores utilizando una dirección relativa a una base más un índice.

**Conclusión:** Durante este taller, aprendimos sobre los diferentes modos de direccionamiento en el ensamblador x86. Cada modo tiene sus propias ventajas y desventajas. El direccionamiento directo es simple y directo, pero puede ser menos flexible en comparación con otros modos. El direccionamiento relativo es útil cuando se trabaja con estructuras de datos dinámicas, ya que permite acceder a los datos relativos a una posición de memoria base. El direccionamiento por registro es rápido y eficiente, pero puede ser limitado en la cantidad de datos que se pueden manipular al mismo tiempo. En general, la elección del modo de direccionamiento depende de las necesidades específicas del programa y de la eficiencia requerida. Es importante entender los diferentes modos de direccionamiento para poder escribir código eficiente y optimizado en ensamblador.