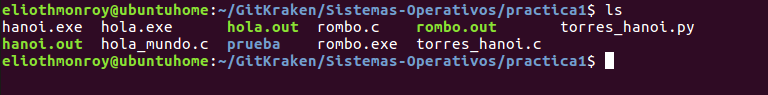
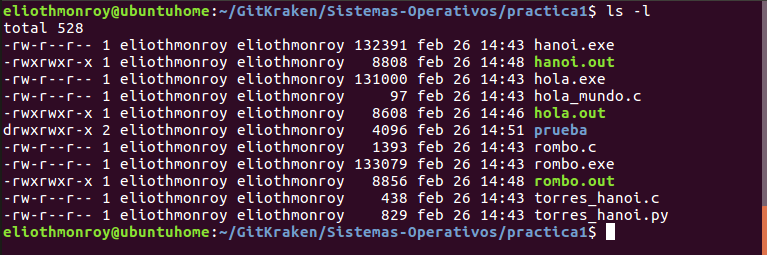
# Linux

## 5

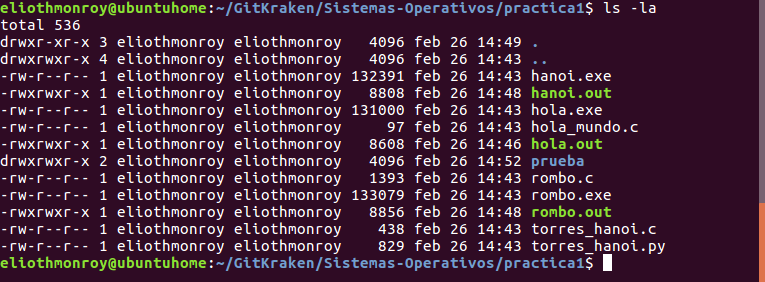
### ls



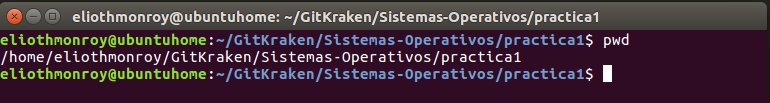
### ls -l



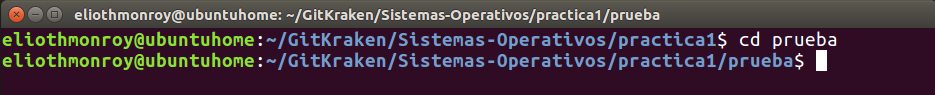
### ls -la



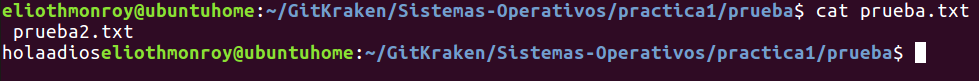
### pwd



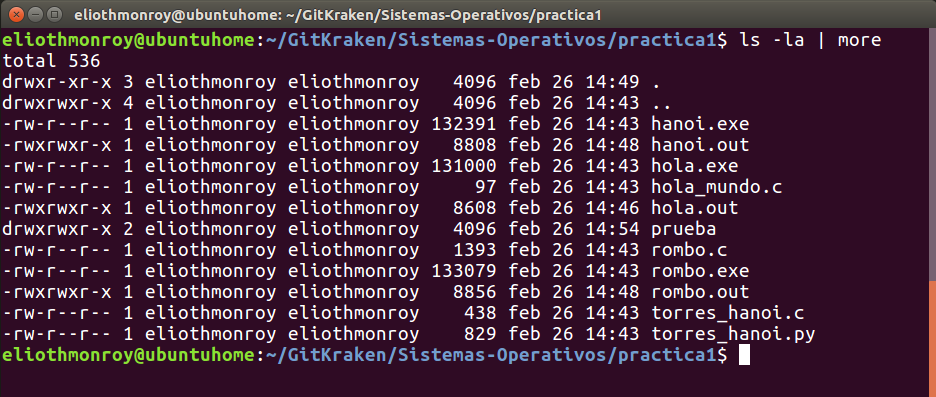
### cd



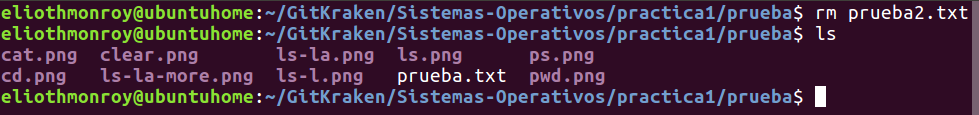
### cat



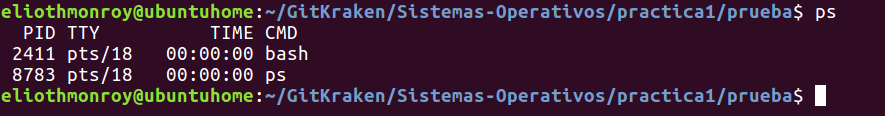
### ls -la|more



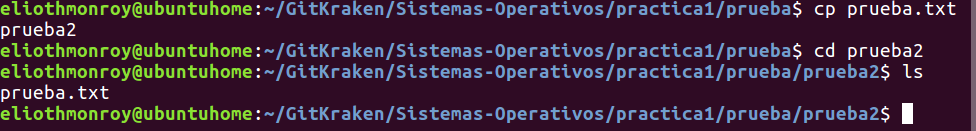
### rm



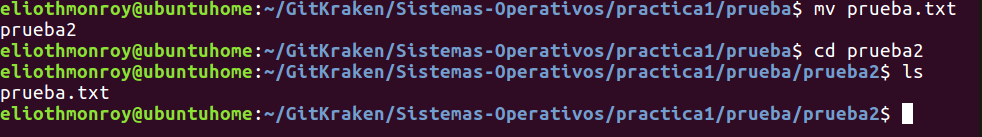
### ps



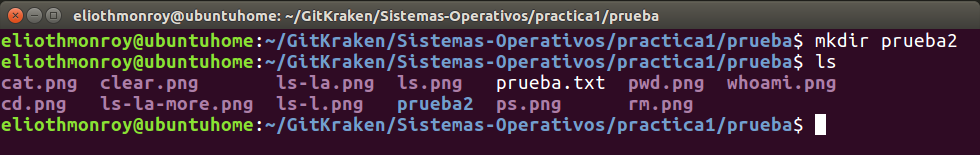
### cp



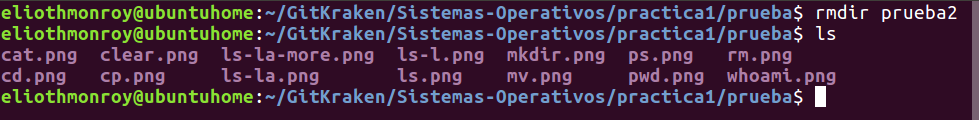
### mv



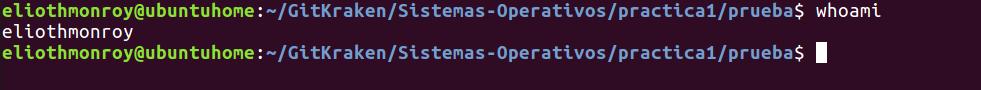
### mkdir



### rmdir



### whoami



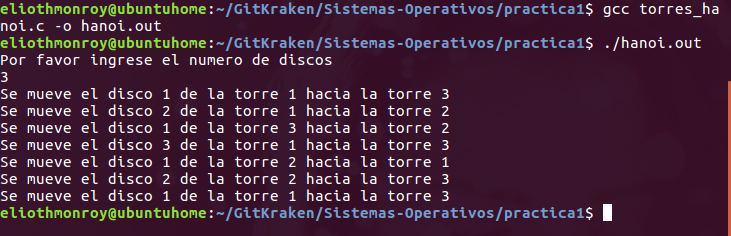
## 9

## Torres de hanoi

Código torres\_hanoi.c

|  |
| --- |
| #include<stdio.h>  **void** hanoi(**int** numero\_discos, **int** inicio, **int** final){  **if**(numero\_discos){  hanoi(numero\_discos-1, inicio, 6-inicio-final);  printf("Se mueve el disco %d de la torre %d hacia la torre %d**\n**", numero\_discos, inicio, final);  hanoi(numero\_discos-1,6-inicio-final,final);  }  }  **int** main(**int** argc, **char** **const** \*argv[])  {  hanoi(4,1,3);  **return** 0;  } |

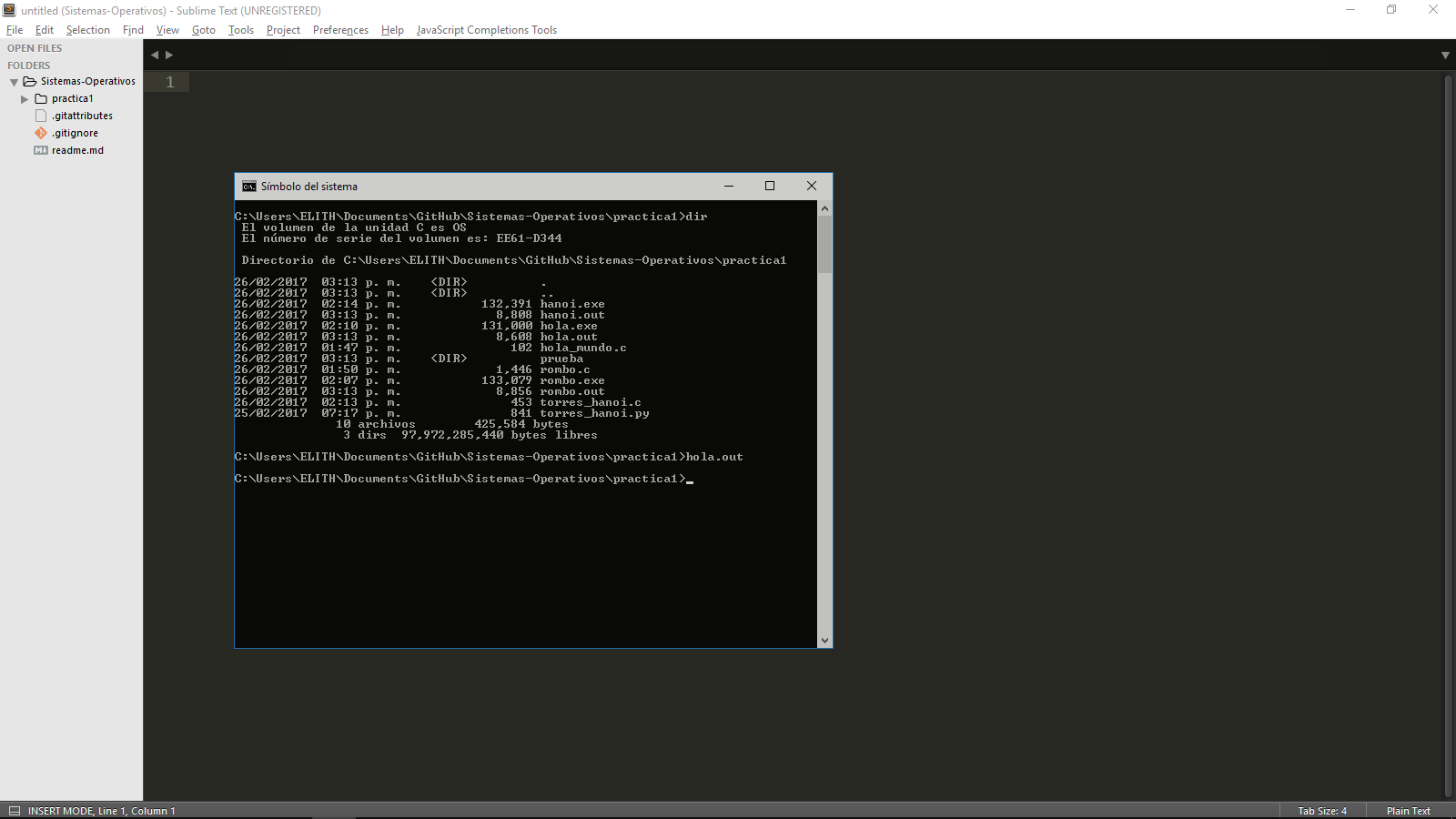
Compilación y ejecución:



# Windows

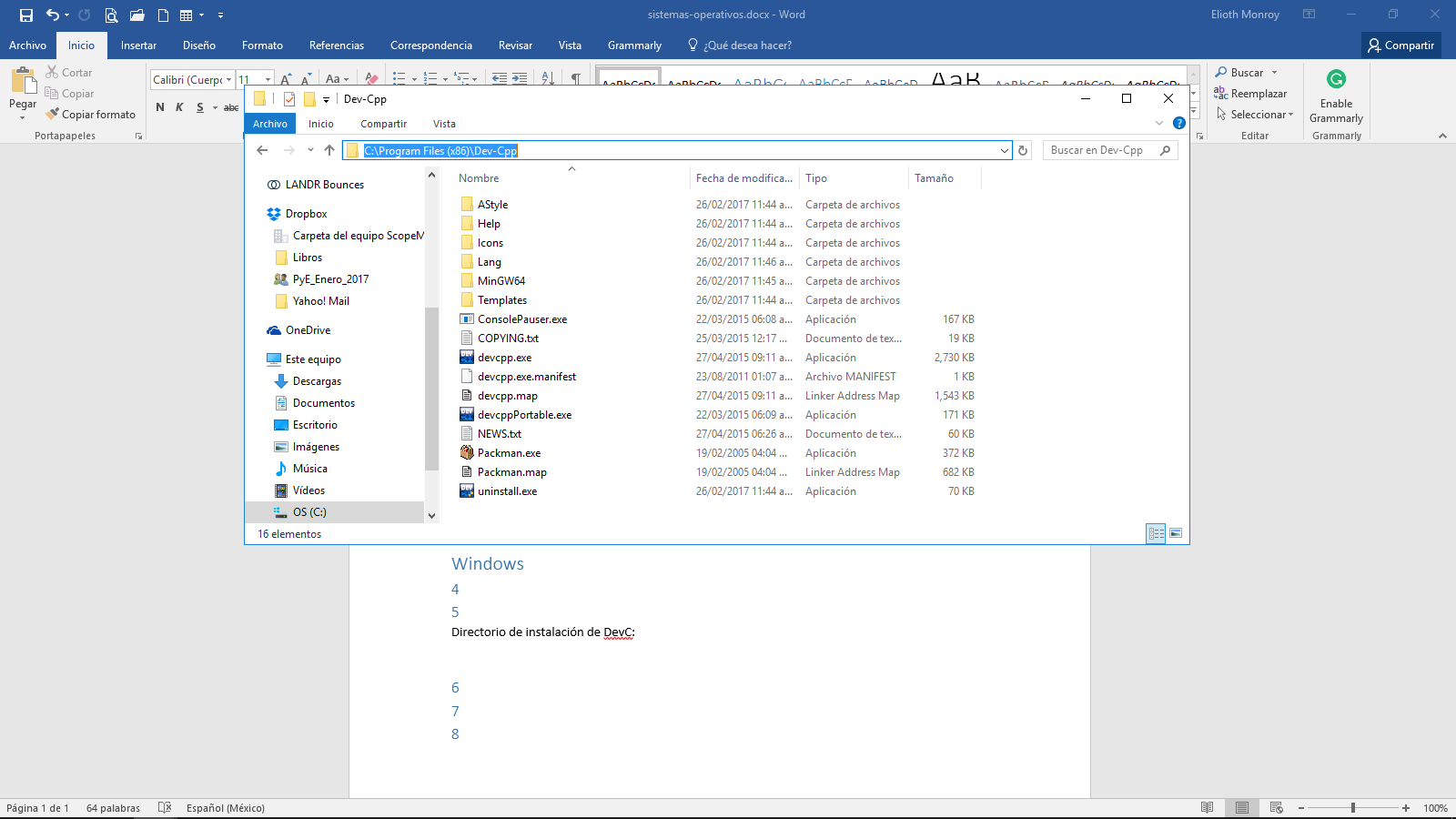
## 4

Al intentar ejecutar alguno de los archivos .out, Windows no reconoce la instrucción, e intenta abrir el archivo con algún programa que se tenga instalado, en caso de no encontrar alguno pide al usuario que indique como desea abrir el archivo, si no se indica no produce ninguna salida la ejecución de los programas.



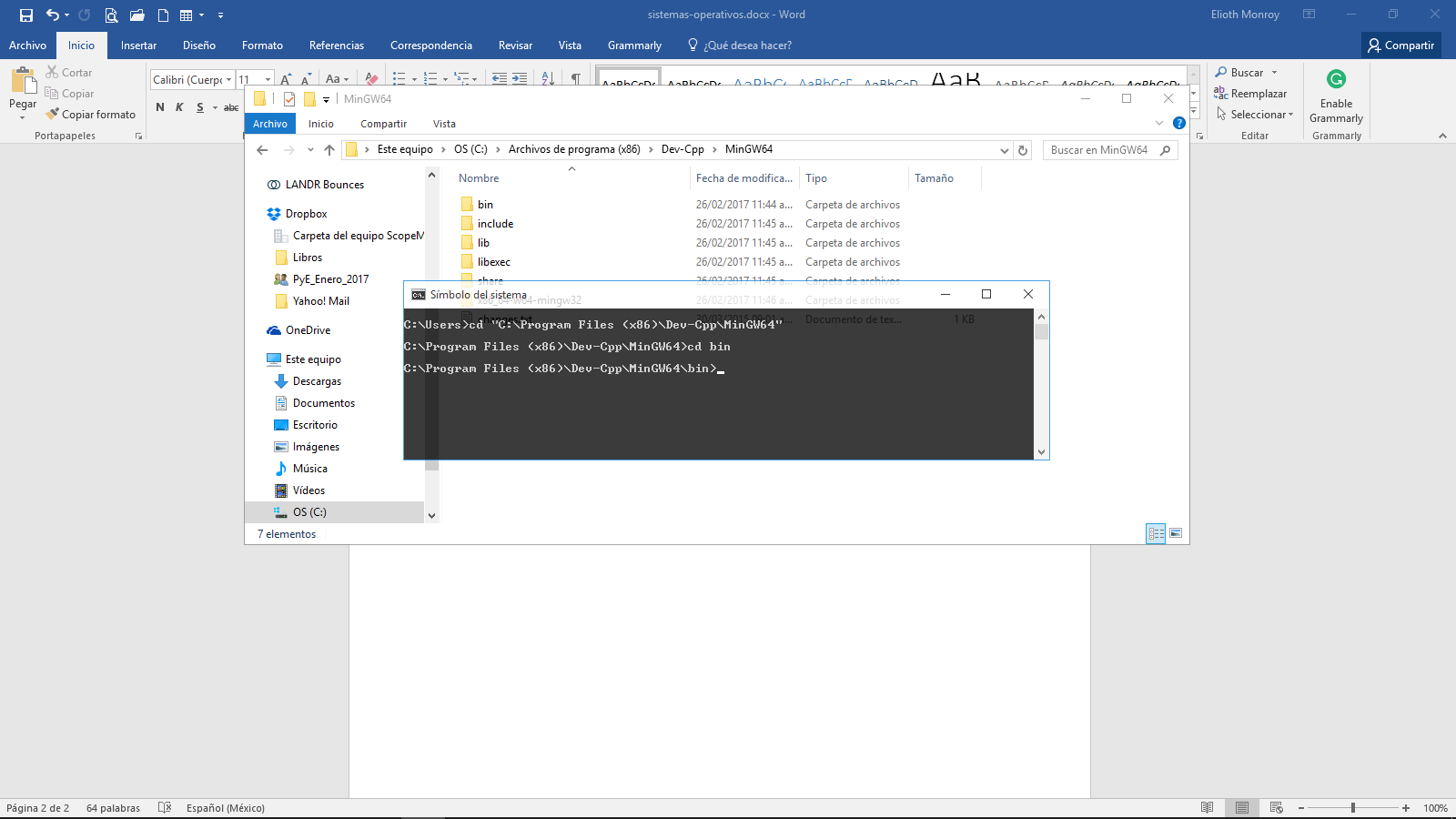
## 5

Directorio de instalación de DevC:



## 6

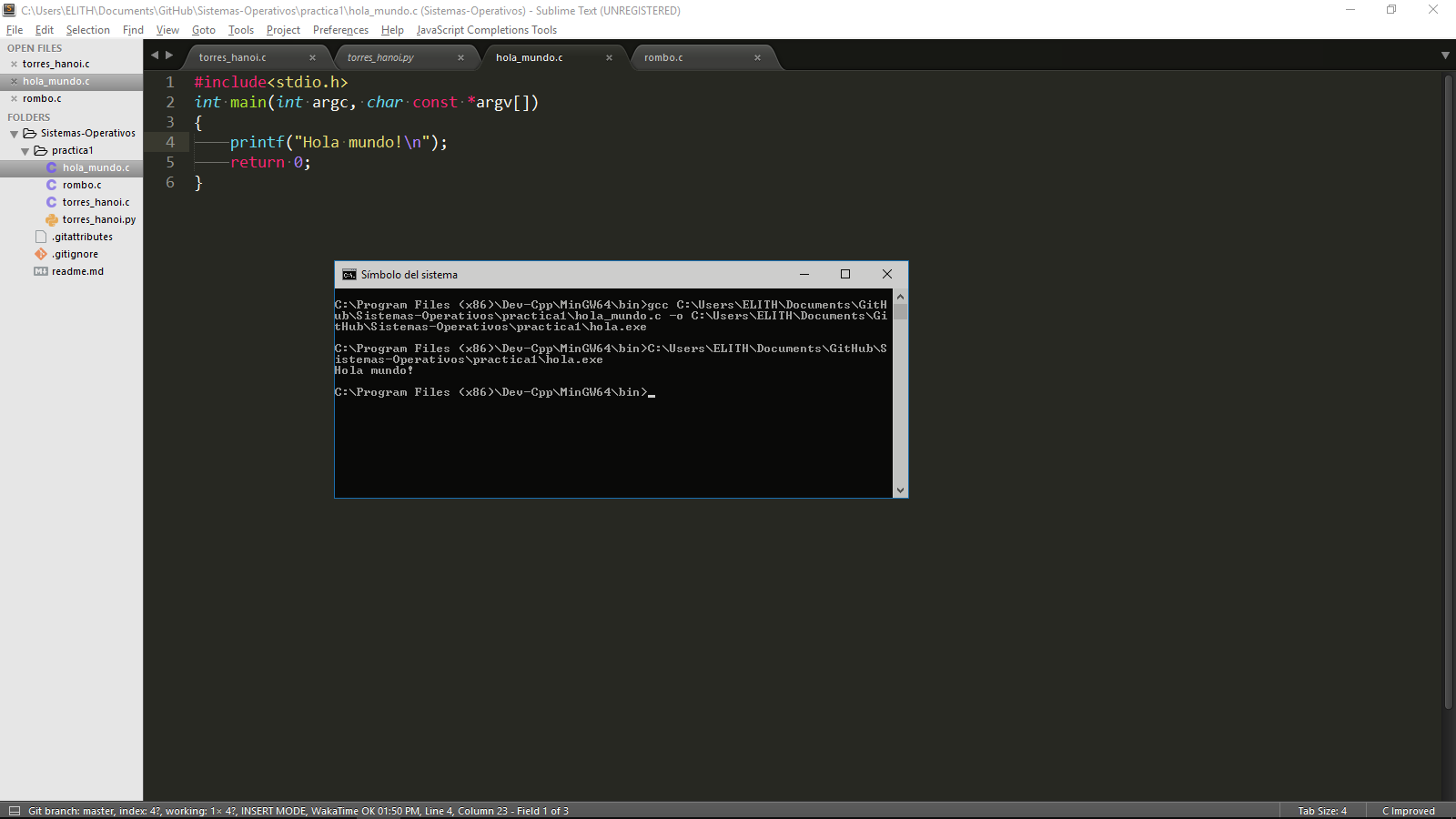
Después de encontrar el folder de instalación de DevC, accedemos a el usando el comando cd y posteriormente nos ubicamos en la carpeta bin donde se encuentra el compilador gcc para c.



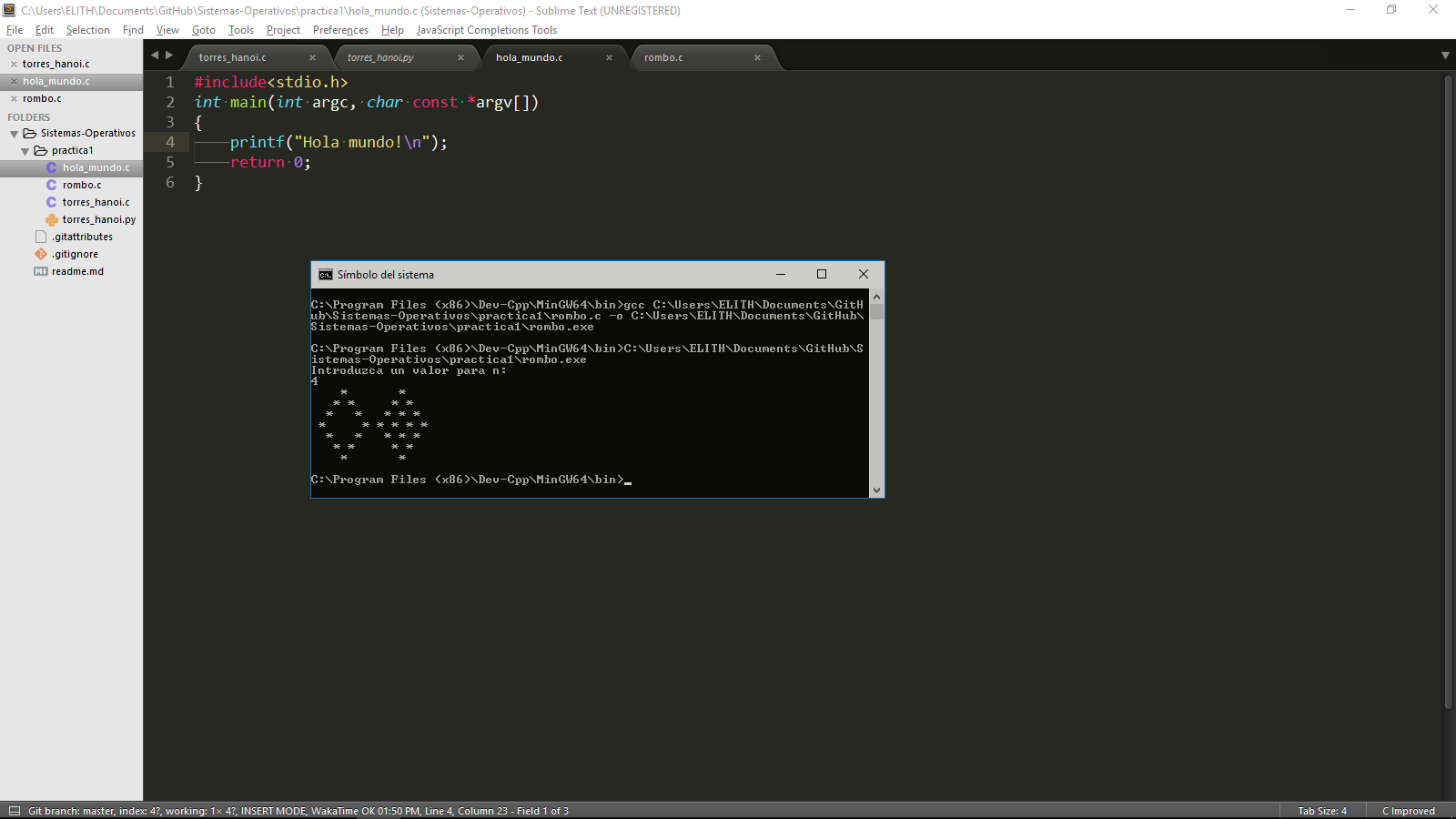
## 7

Ahora procedemos a ejecutar cada uno de los programas que realizamos anteriormente en Linux, pero ahora desde el CMD de Windows.

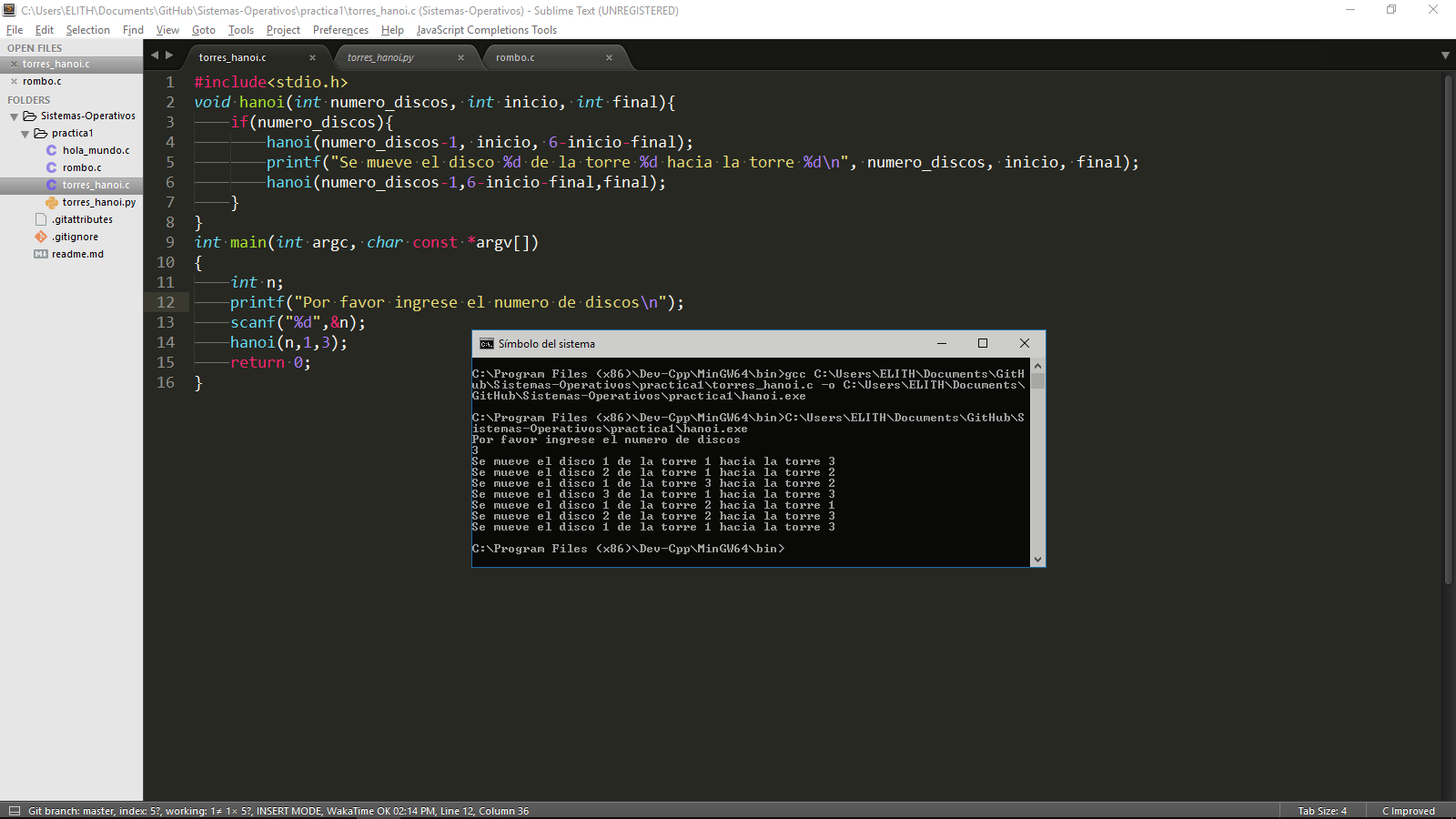
### Hola Mundo



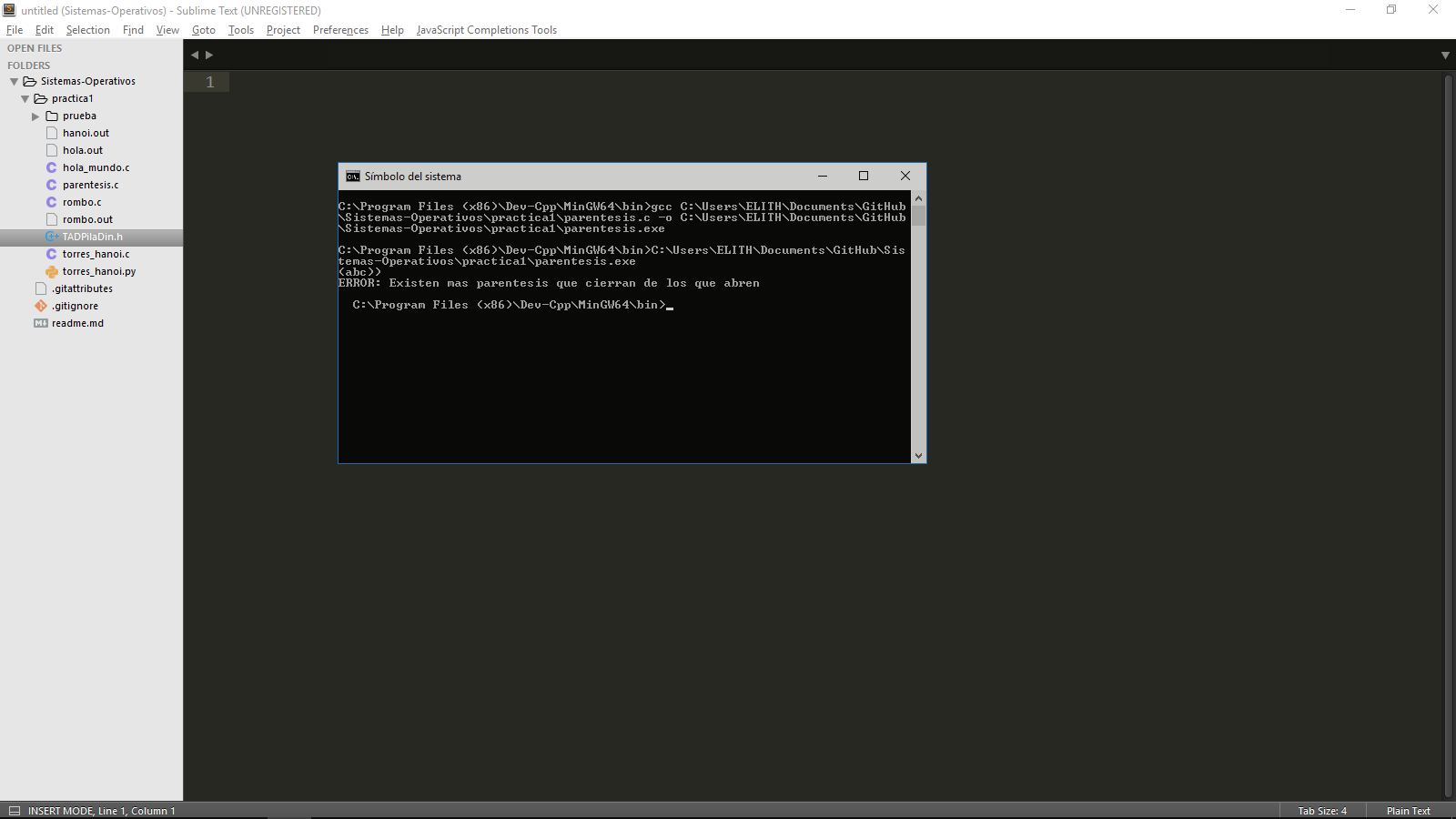
### Rombos



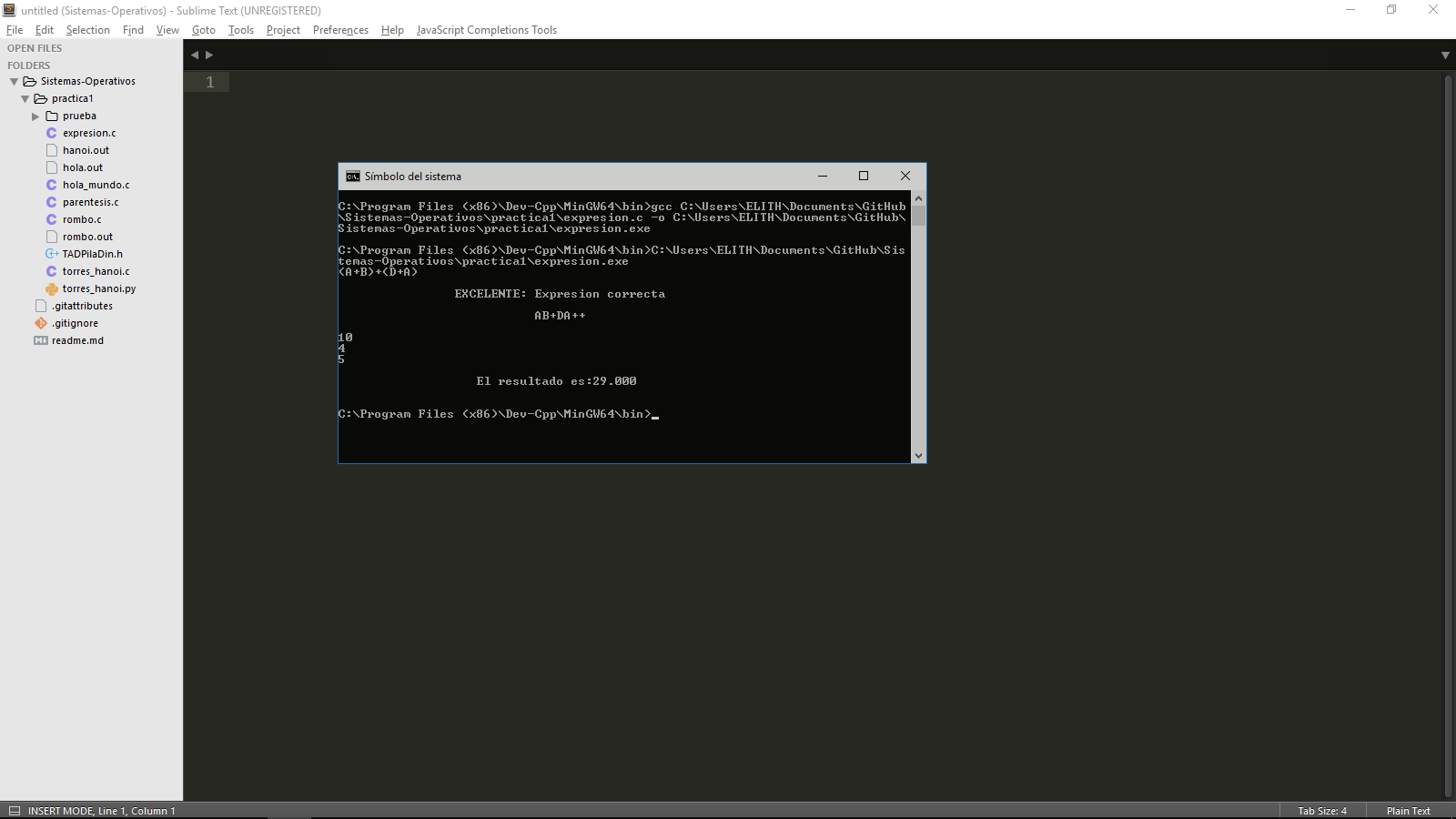
### Torres de Hanoi



### Paréntesis Balanceados



### Expresiones Aritméticas



## 8

Ahora que hemos trabajado tanto en el entorno Linux como en Windows, podemos reportar algunas diferencias y similitudes entre los comandos disponibles en cada uno, así como la forma de compilación y ejecución de los programas escritos en lenguaje C.

|  |  |
| --- | --- |
| **Similitudes** | **Diferencias** |
| * Podemos realizar las mismas acciones en ambos sistemas operativos. * En ambos entornos es posible trabajar con programas hechos en C, mientras se tenga disponible un compilador. * La forma de compilar programas C es igual para ambos sistemas operativos, lo que cambia es el tipo de archivo que necesitamos para ejecutar nuestros programas. * El compilador en ambos entornos nos genera un ejecutable valido dependiendo del sistema operativo. * Si se omite el nombre del archivo de salida al compilar, en ambos entornos el compilador lo nombra como a.out o a.exe en Linux y Windows respectivamente. | * Los nombres para algunos comandos son distintos dependiendo del sistema operativo (por ejemplo: ls y dir). * Linux nos da una mayor cantidad de herramientas para trabajar desde la línea de comandos, debido a que está orientado para usuarios especializados. * Linux incluye de forma nativa el compilador GCC para trabajar con C y C++, mientras que en Windows es necesario instalarlo. * En Linux al compilar un programa escrito en c, este nos genera un ejecutable con extensión .out. Mientras que, en Windows, el compilador nos genera un archivo ejecutable .exe. |