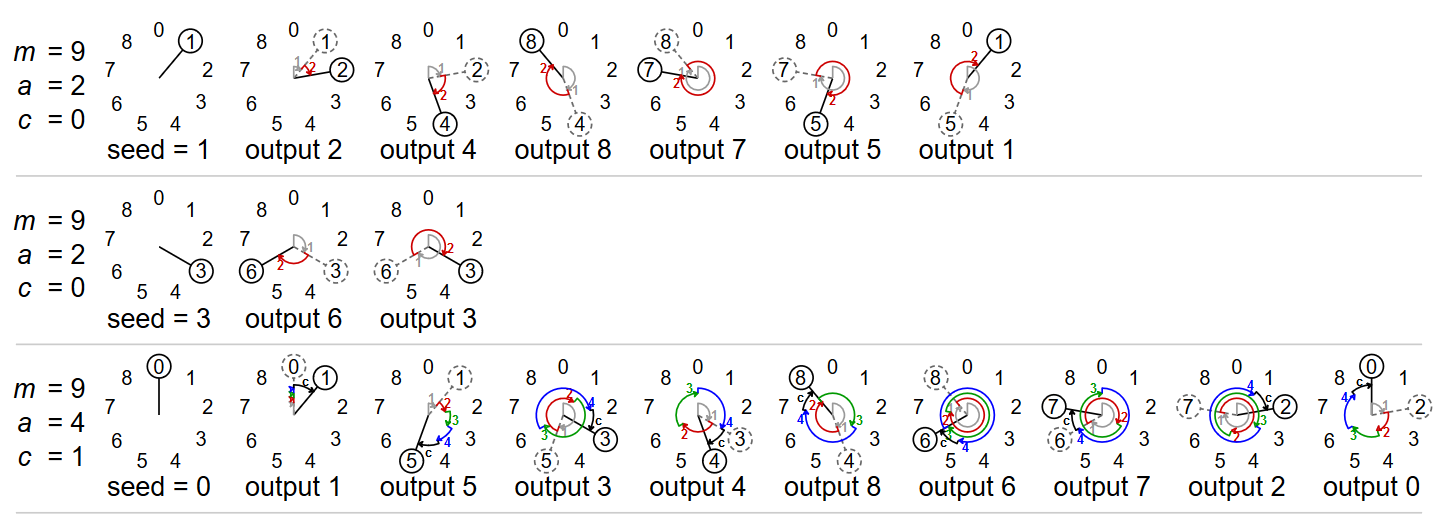
INFORME INSTRUCTIVO:



A grandes rasgos, lo que hace mi programa es muy simple, genera un numero semi aleatorio, siguiendo la formula recurrente del LCG (Linear Congruential Generator); primero pide un valor para la “semilla” que es el numero inicial “Xn”, y luego un segundo valor para definir cuantas veces debe ejecutarse la formula para generar el número (si se ingresa un “0”, por razones obvias se va a imprimir la semilla). Después de que el número es generado, se pide ingresar otro número por teclado para intentar adivinar que numero se generó con el LCG, si fallas, el programa te dará una pista si el numero generado es mayor o menor.

El programa está estructurado de la siguiente forma:

Primero están definidos los mensajes que se imprimen junto a todas las variables, incluidas las que se van a usar en la formula, después se capturan los números ingresados por el teclado, se guardan en sus respectivas variables y luego se ejecuta el ciclo recursivo, la formula del LGC, anoté varios comentarios en el programa los detalles sobre exactamente qué operación hace cada línea. Después de terminar el ciclo del LGC, se imprime el numero aleatorio para poder ver si el algoritmo funciona correctamente, esta sección del código está marcada (;desde aquí, ;hasta aquí) por si después alguno quiere transformar ese trozo del código en comentarios, y jugar un rato sin saber cual es el numero generado. Después vamos al segundo ciclo, que es la parte donde se captura el numero ingresado para adivinar el numero generado, se hace una comparación y usando los [saltos condicionales](https://www.tutorialspoint.com/assembly_programming/assembly_conditions.htm), el programa muestra un mensaje u otro; si no adivinas correctamente, se devuelve al principio del ciclo y captura el siguiente numero ingresado, para hacer nuevamente la comparación, hasta que el usuario adivine o cierre el programa.

Ejemplo de cómo funciona el algoritmo:

*\*Las imágenes ilustrativas son cortesía de* [*Wikipedia*](https://en.wikipedia.org/wiki/Linear_congruential_generator)