

ANÁLISIS

- Dada una cadena, obtener su código Ascii y mostrar su equivalente en binario.

OBJETIVO

- Mostrar el código binario de una cadena, la cual es proporcionada por el usuario.

ENTRADA

- sLetter[50]: variable de tipo string que almacenará el mensaje de la cadena dada por el usuario. El 50 significa que puede guardar hasta esa cantidad de caracteres.

PRE-CONDICIONES

- eCount será una variable contador de tipo entero que irá aumentando de uno en uno.
- eReverse será una variable de tipo entero que irá disminuyendo su hasta que queden ordenados los números según el código binario.
- eResult[8] será un arreglo de tipo entero que almacenará el módulo del número ascii dividido entre 2.
- A pesar de que la variable de entrada sLetter está declarada como un string, sólo recibe un carácter.

RESTRICCIONES

- Sólo se aceptará un carácter.

SALIDA

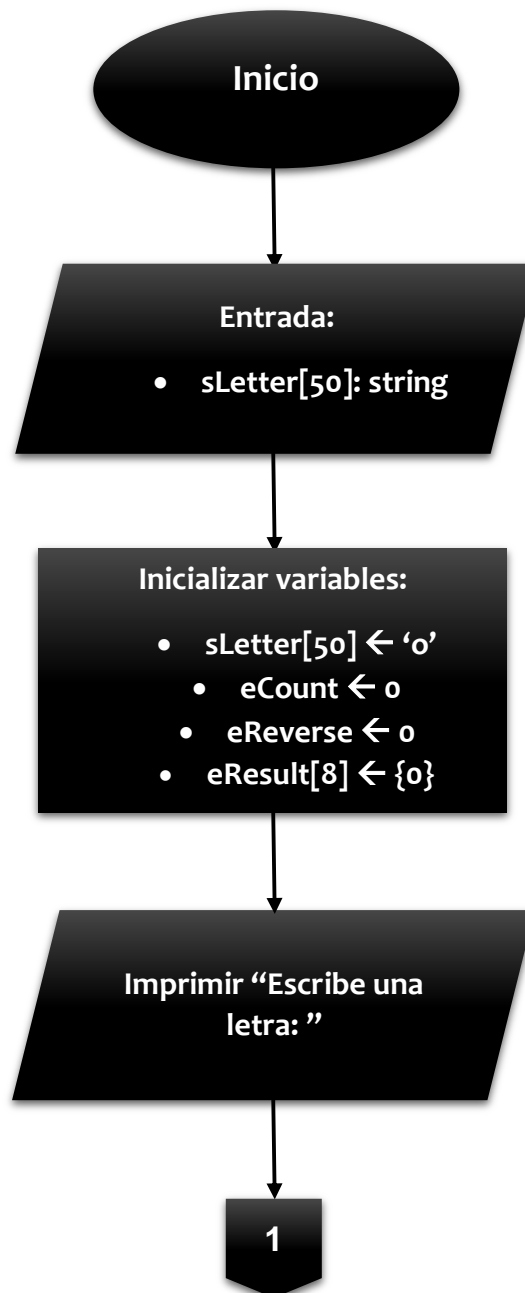
- Imprimir eResult[eReverse]

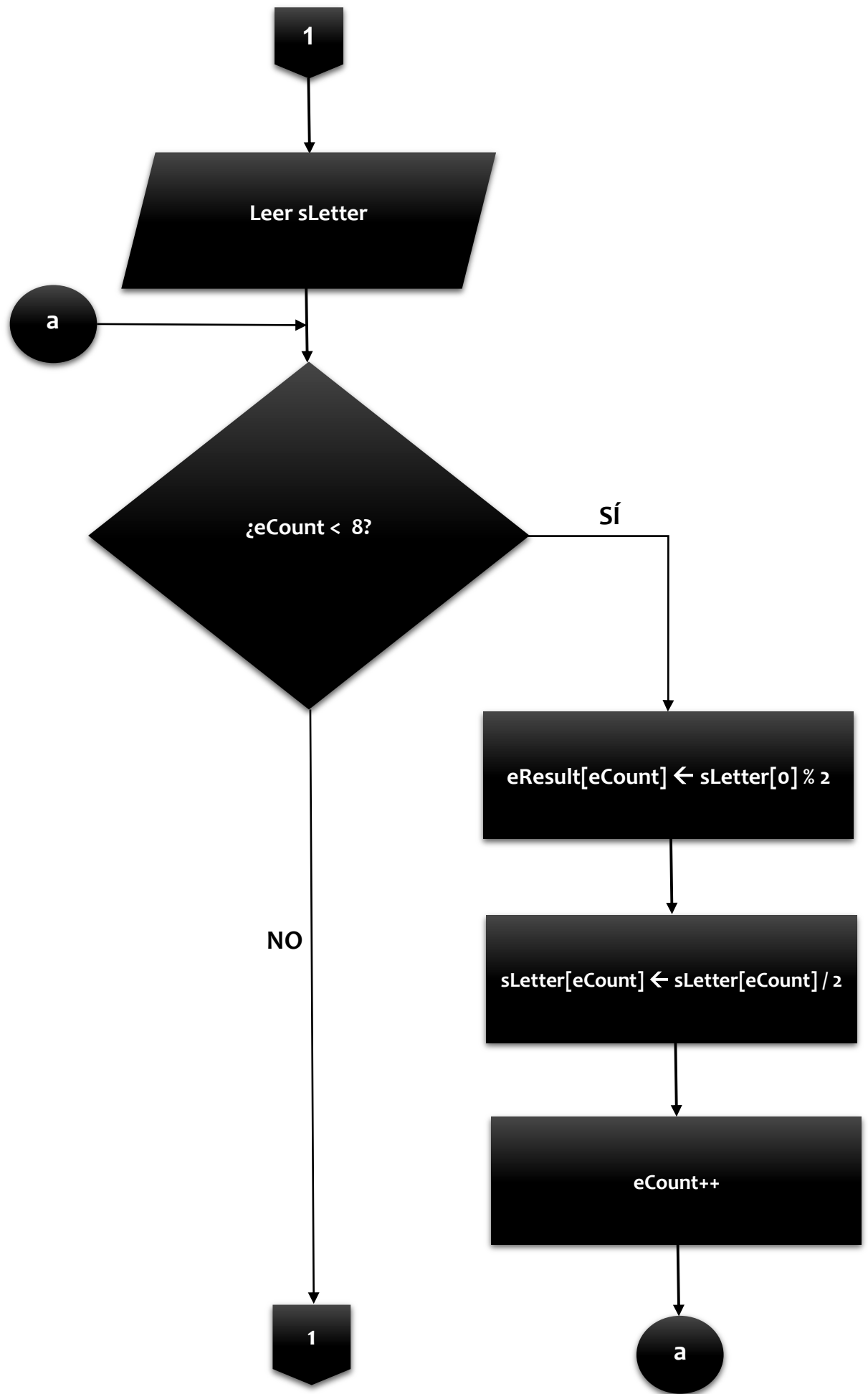
PSEUDOCÓDIGO

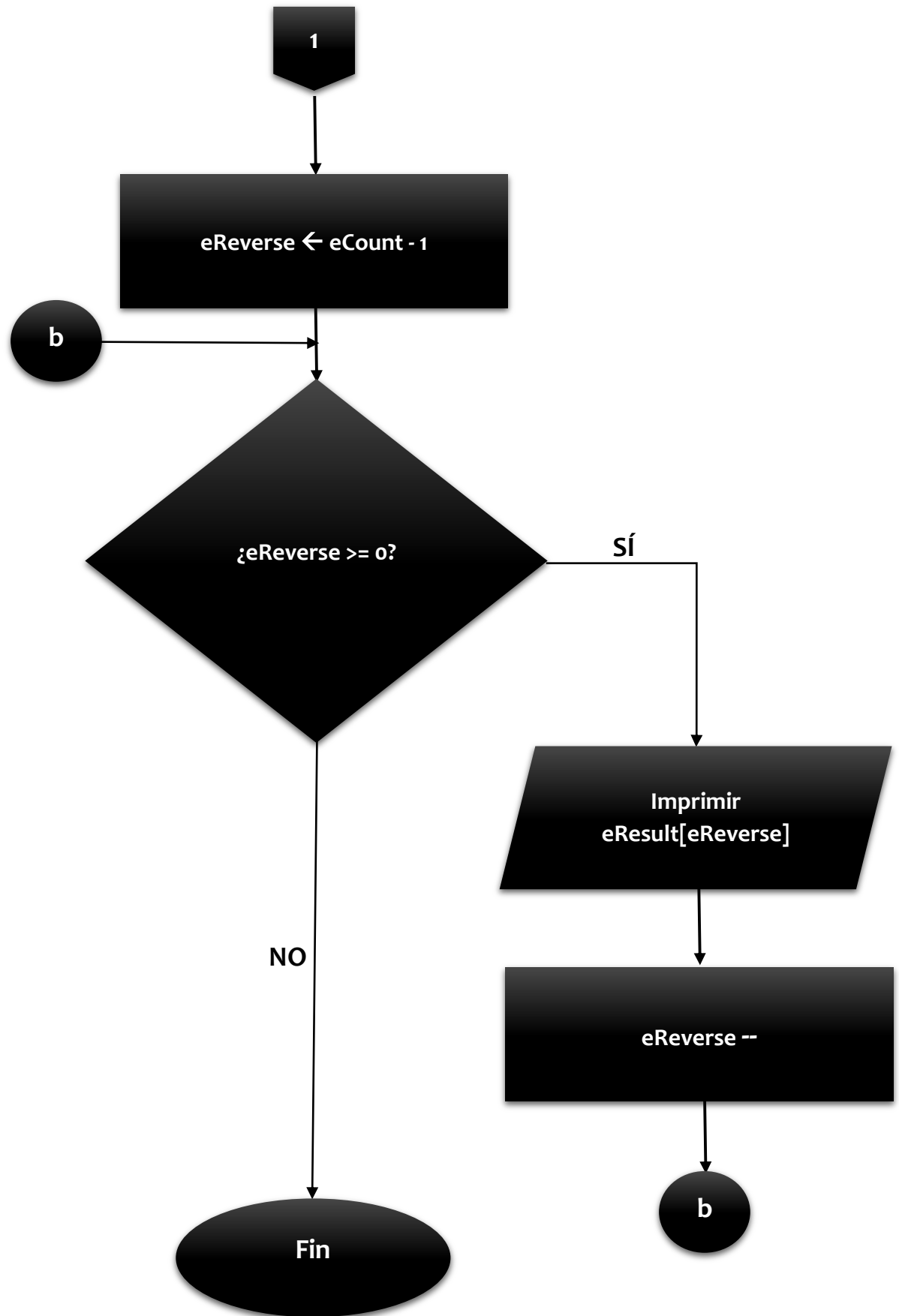
0. Inicio
1. Inicializar variables:
 - 1.1 sLetter[50] \leftarrow ""
 - 1.2 eCount \leftarrow 0
 - 1.3 eReverse \leftarrow 0
 - 1.4 eResult[8] \leftarrow {0}
2. Imprimir "Escribe una letra: "

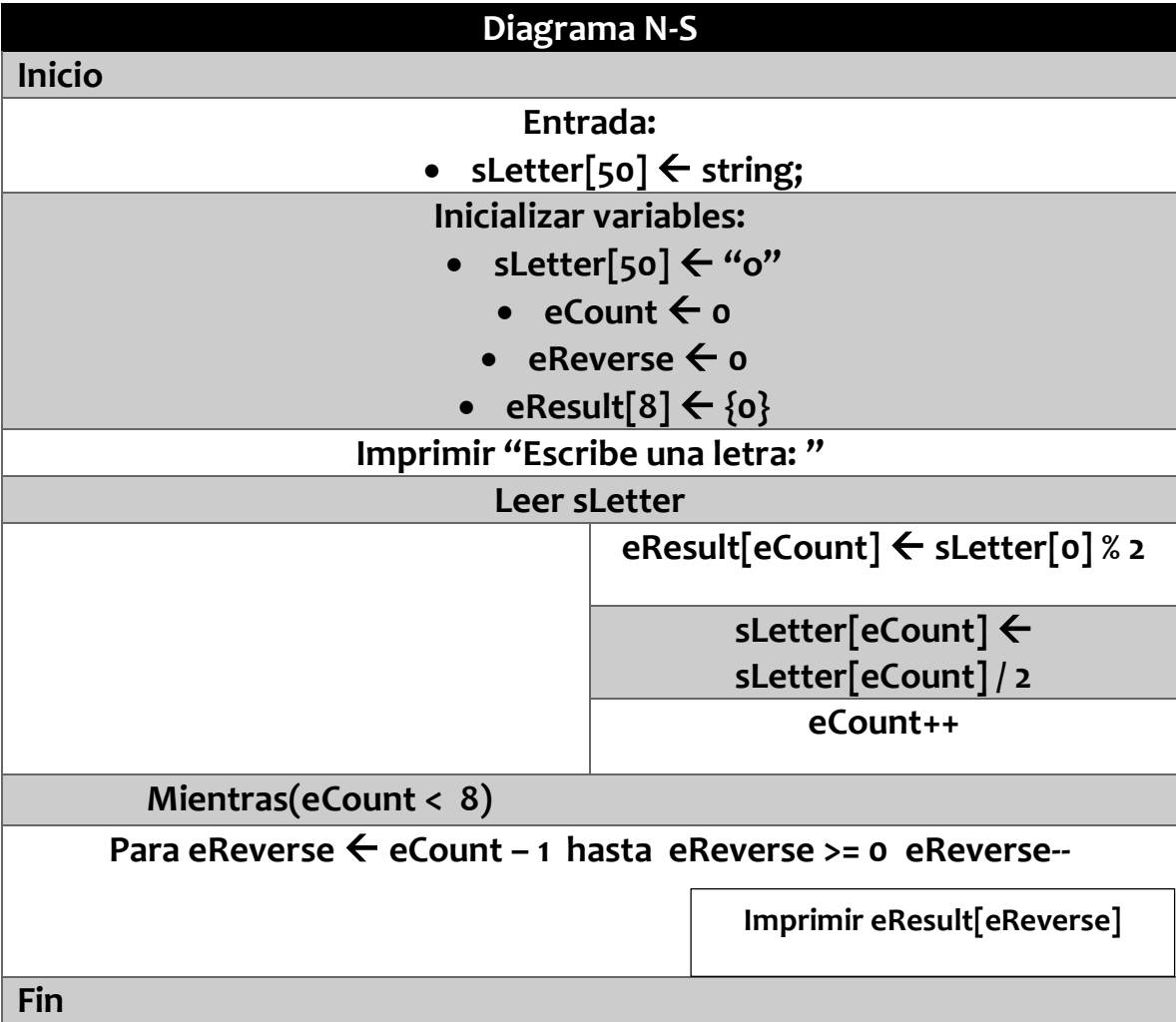
3. Leer sLetter
4. Mientras(eCount < 8)
 - 4.1 eResult[eCount] \leftarrow sLetter[0] % 2
 - 4.2 sLetter[eCount] \leftarrow sLetter[eCount] / 2
 - 4.3 eCount++
5. Fin Mientras
6. Para(eReverse = eCount - 1; eReverse >= 0; eReverse--)
 - 6.1 Imprimir eResult[eReverse]
7. Fin Para
8. Fin.

DIAGRAMA DE FLUJO









Prueba de escritorio		
Prueba	Datos de entrada	Salida
	sLetter	eResult[eReverse]
1	h	01101000
2	a	01100001
3	l	01101100
4	o	01101111