

CERO UNO

I. ANÁLISIS

Descripción del problema: Dada una cadena compuesta de ceros y unos, se busca comprobar si esta cadena está formada por una cantidad n de ceros seguida de una cantidad n de unos.

Datos de entrada: cadena

Datos de salida: 0, 1

II. PSEUDOCÓDIGO

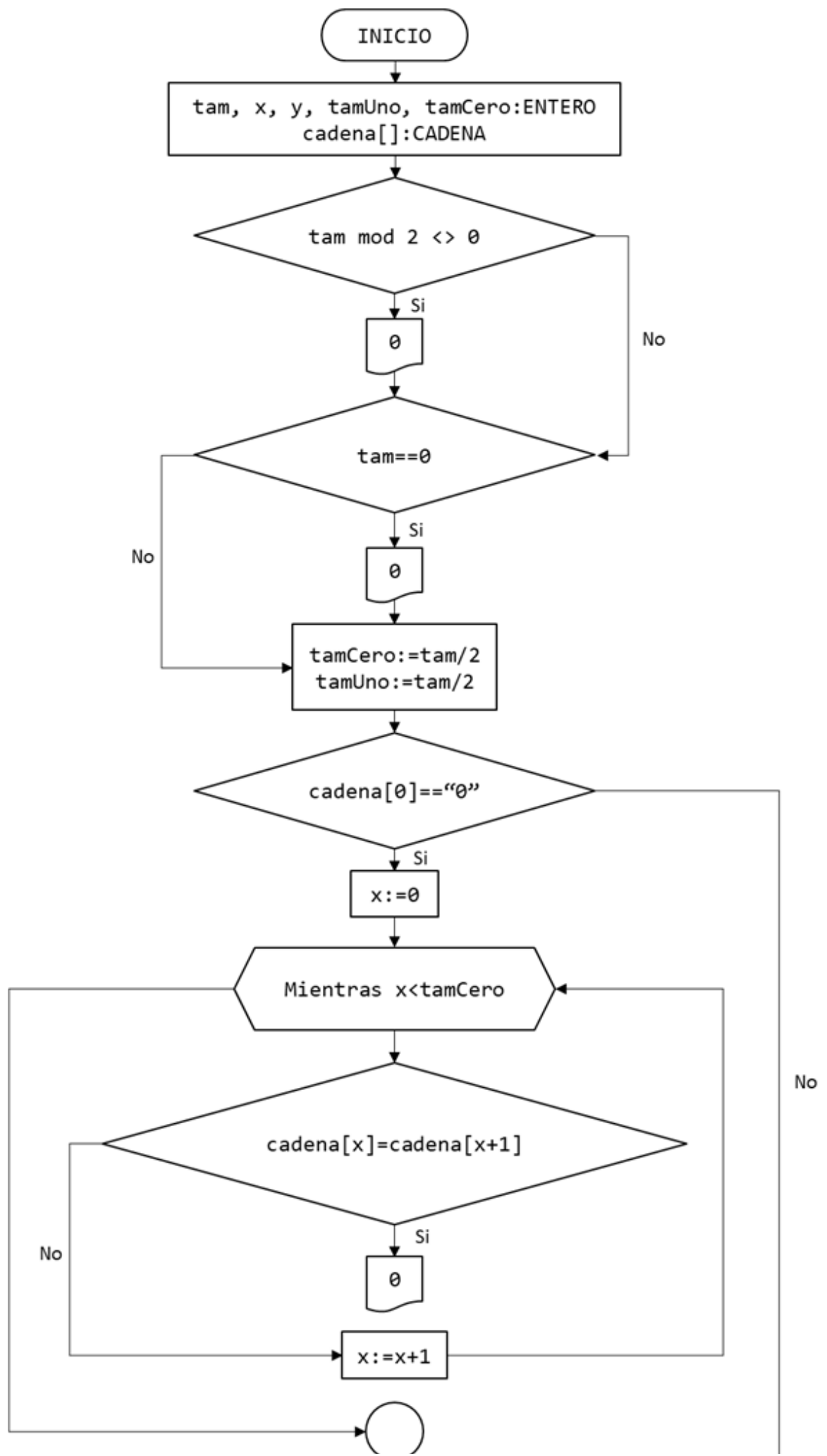
```
INICIO
tam: ENTERO
x: ENTERO
y: ENTERO
cadena[]: CADENA
tam:= STRLEN(cadena);
SI tam %2 <> 0 ENTONCES
    ESCRIBIR "0"
FIN SI
SI tam=0 ENTONCES
    ESCRIBIR "0"
FIN SI
tamCero:= tam/2: ENTERO
tamUno:= tam/2: ENTERO
SI cadena[0] = "0" ENTONCES
    x:=0
    MIENTRAS x< tamCero-1;
        SI cadena[x] <> cadena [x+1] ENTONCES
            ESCRIBIR "0"
        FIN SI
        x:=x+1
    FIN MIENTRAS
FIN SI
SI cadena[x+1]= "1" ENTONCES
    y:=x+1
    MIENTRAS j<tam-1
        SI cadena[y] <> cadena[y+1] ENTONCES
```

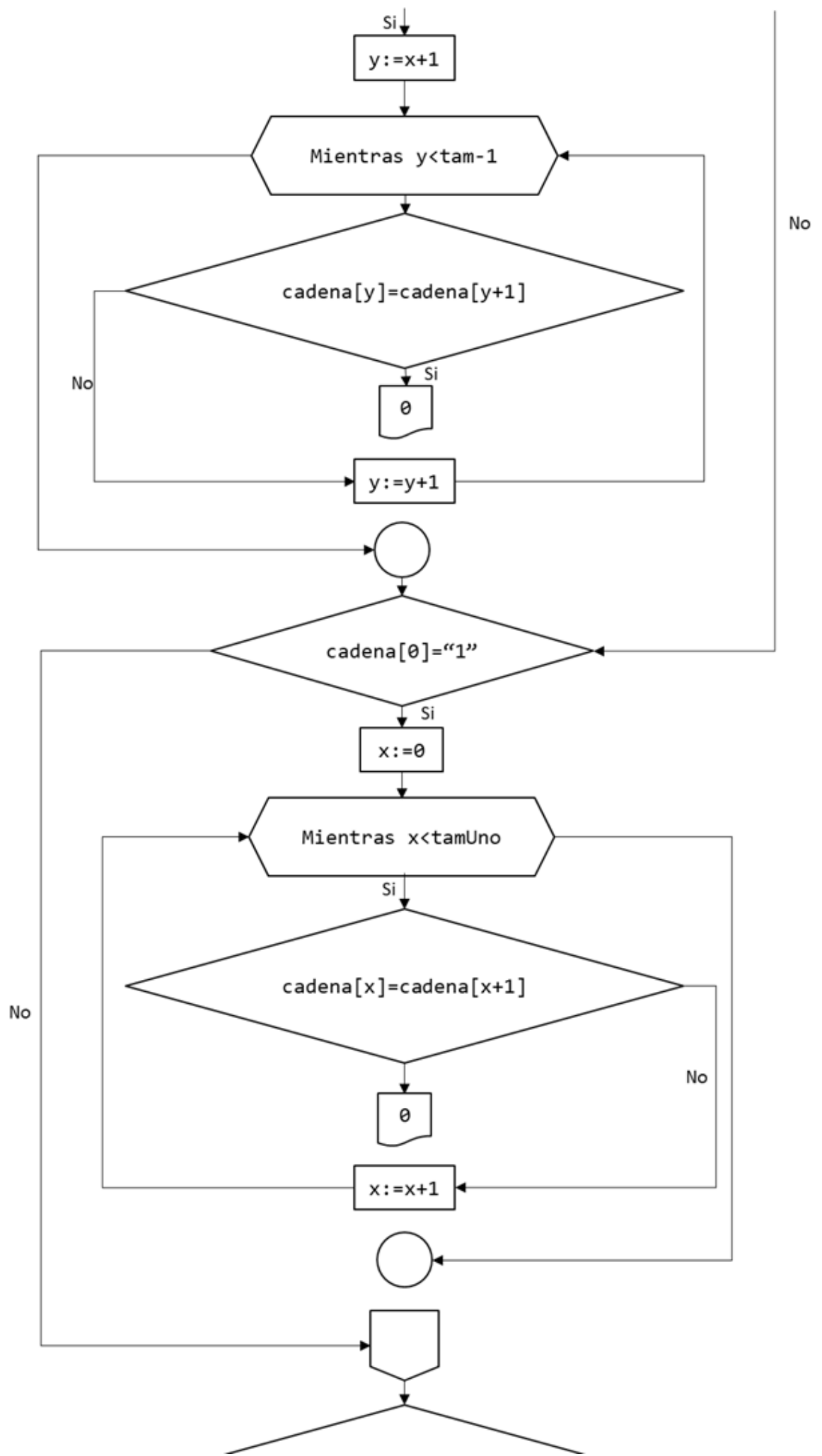
```

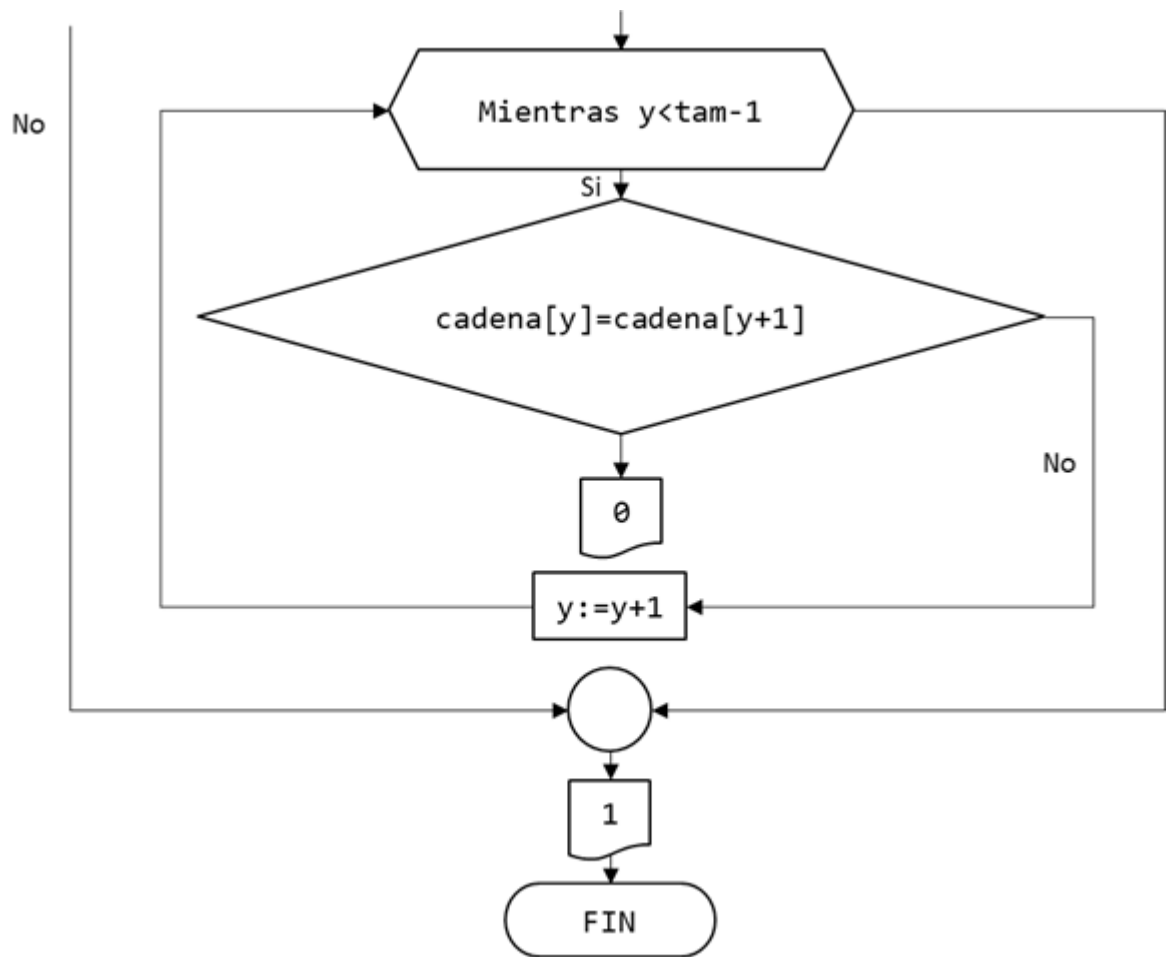
        ESCRIBIR "0"
    FIN SI
    y:=y+1
    FIN MIENTRAS
FIN SI
SI cadena [0]="1" ENTONCES
    x:= 0
    MIENTRAS x< tamUno
        SI cadena[x] <> cadena[x+1] ENTONCES
            ESCRIBIR "0"
        FIN SI
        x:=x+1
    FIN MIENTRAS
FIN SI
SI cadena[x+1]==0 ENTONCES
    y:=x+1
    MIENTRAS y<tam-1; y:= y+1)
        SI cadena[y] != cadena[y+1]
            ESCRIBIR "0"
        FIN SI
        y:=y+1
    FIN MIENTRAS
FIN SI
ESCRIBIR "1"
FIN

```

DIAGRAMA DE FLUJO







III. PRUEBA DE ESCRITORIO

Entrada			Salida
tam	2		0
tamCero	1		
tamUno	1		
x	0	1	
y	2		
cadena	10		
cadena[0]	1		
cadena[1]	0		

IV. CODIFICACIÓN

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int validaCadena(const char* cadena) {
    int tam = strlen(cadena);
    int tcero, tUno, x, y;
    if (tam % 2 != 0) {
        printf("0");
        return 0;
    }

    tcero = tam / 2;
    tUno = tam / 2;

    if (cadena[0] == '"0"') {
        x = 0;
        for (x = 0; x < tcero - 1; x++) {
            if (cadena[x] != cadena[x + 1]) {
                printf("0");
                return 0;
            }
        }
        if (cadena[x + 1] == '"1"') {
            for (y = x + 1; y < tam - 1; y++) {
                if (cadena[y] != cadena[y + 1]) {
                    printf("0");
                    return 0;
                }
            }
        }
    }
    else if (cadena[0] == '"1"') {
        for (x = 0; x < tUno - 1; x++) {
            if (cadena[x] != cadena[x + 1]) {
                printf("0");
                return 0;
            }
        }
    }
}
```

```
        }  
    }  
}  
printf("1");  
return 0;  
}  
int main(){  
    char* s;  
    printf("Ingrese una cadena:\n");  
    s = malloc (256);  
    scanf("%256s", s);  
    validaCadena(s);  
    return 0;  
}
```