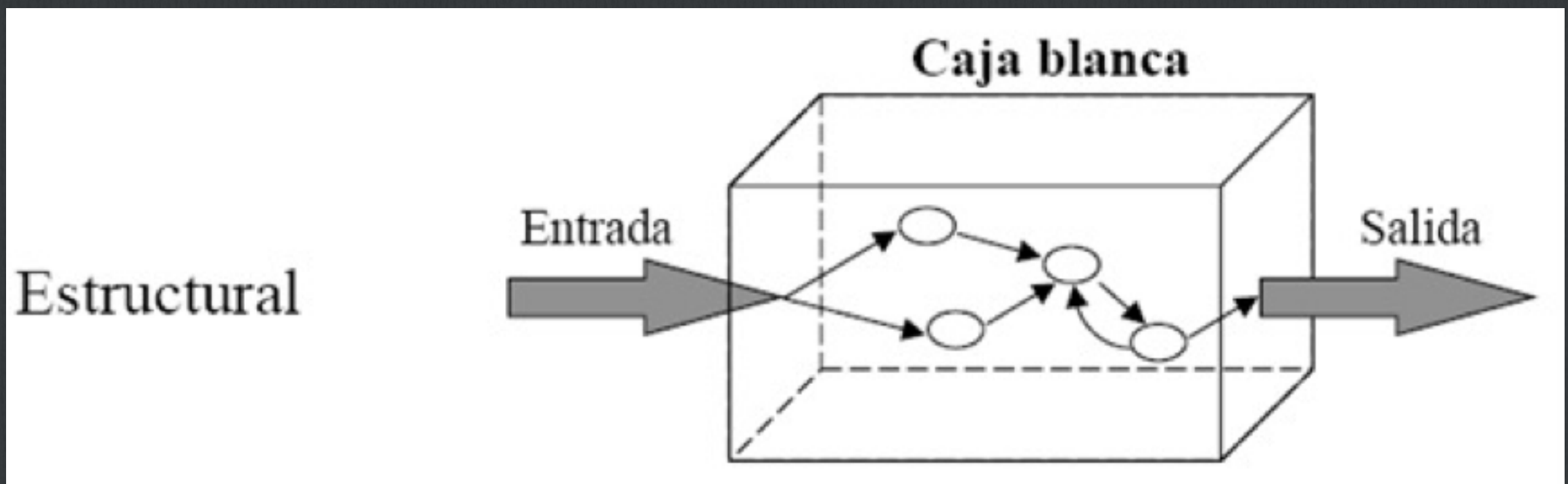


PRUEBAS DE CAJA BLANCA

UT 3 : DISEÑO Y REALIZACIÓN DE PRUEBAS

DEFINICIÓN (I)



También se las conoce como pruebas estructurales o de caja de cristal. Se basan en el examen minucioso de los detalles del código de nuestra aplicación.

DEFINICIÓN (II)

Mediante esta técnica se pueden obtener casos de prueba que:

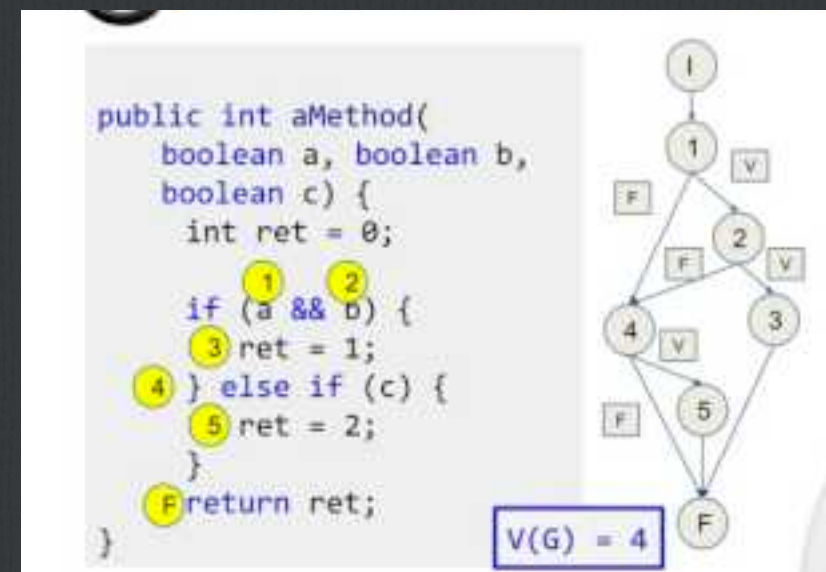
- ☐ **Garanticen que se ejecutan al menos una vez todos los caminos independientes de cada módulo.**
- ☐ **Ejecuten todas las sentencias al menos una vez.**
- ☐ **Ejecuten todas las decisiones lógicas en su parte verdadera y en su parte falsa.**
- ☐ **Ejecuten todos los bucles en sus límites.**
- ☐ **Utilicen todas las estructuras de datos internas para asegurar su validez.**

PRUEBA DEL CAMINO BÁSICO

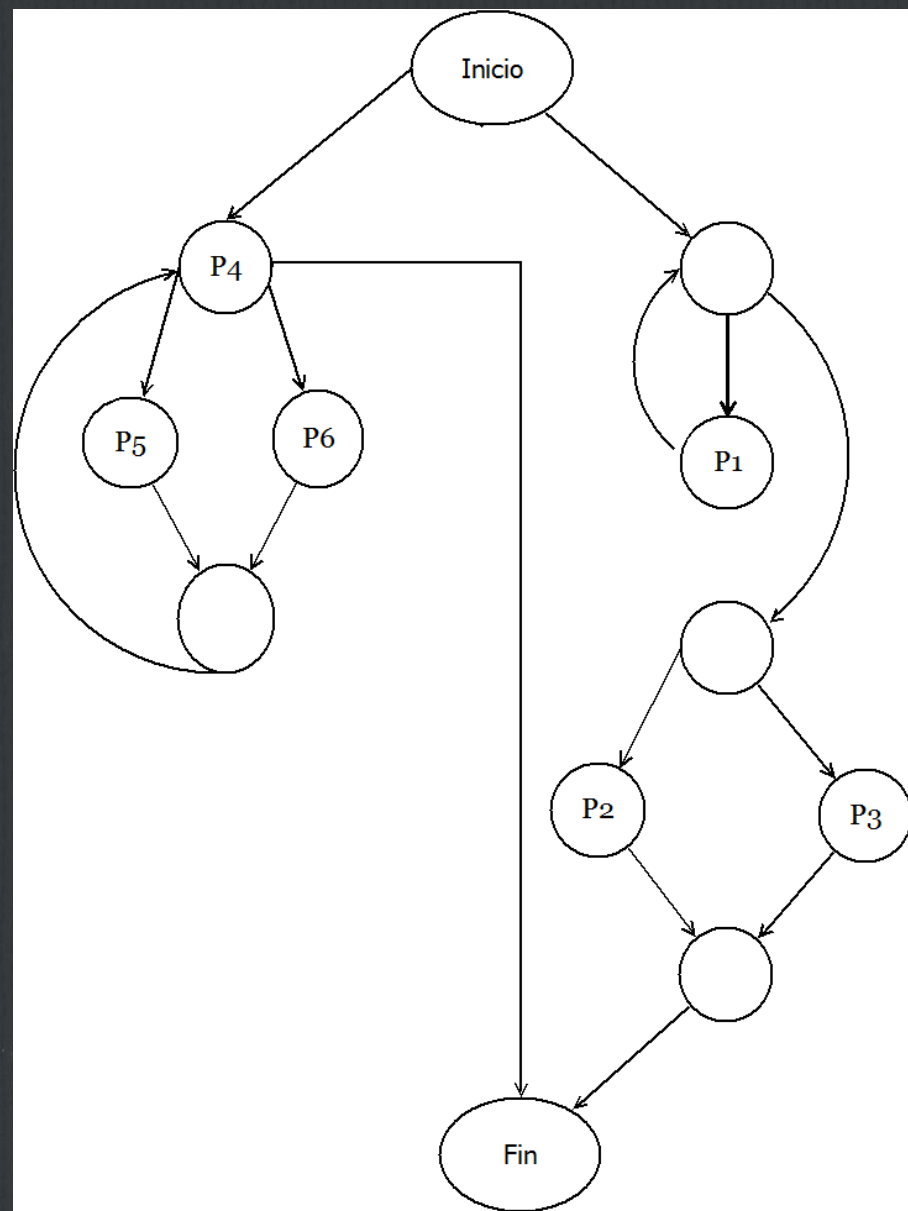
Una de las técnicas utilizadas para desarrollar los casos de prueba de caja blanca es la prueba del camino básico que consiste en verificar el código de nuestros sistemas de manera que comprobemos que todo funciona correctamente, es decir, se debe verificar que todas las instrucciones del programa se ejecutan por lo menos una vez.

Los pasos para desarrollar la prueba del camino básico son:

- 1.- Dibujar el grafo de flujo.
- 2.- Calcular la complejidad ciclomática.
- 3.- Determinar el conjunto básico de caminos independientes.
- 4.- Generar los casos de prueba



PRUEBA DEL CAMINO BÁSICO: GRAFO DE FLUJO.




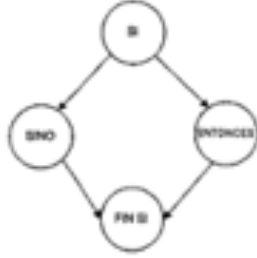

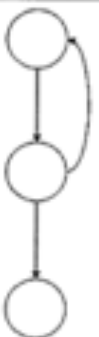
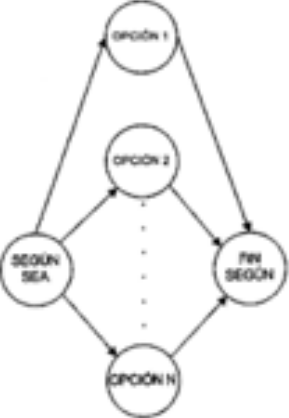
Según Wikipedia, un grafo de flujo (CFG) es una representación, en forma de grafo dirigido, de todos los caminos que pueden ser atravesados a través de un programa durante su ejecución.

Cada nodo del grafo representa un bloque contiguo de código sin bifurcaciones, conocidos como bloque básico.

Existen dos bloques especiales, el de entrada, que ninguna arista llega a él y es la única forma de empezar a recorrer el grafo, y el de salida, del que ninguna arista sale y es la única forma de terminar de recorrer el grafo.

PRUEBA DEL CAMINO

BÁSICO: GRAFO DE FLUJO (II)

| ESTRUCTURA | GRAFO DE FLUJO |
|--|---|
| SECUENCIAL Instrucción 1 Instrucción 2 Instrucción n |  |
| CONDICIONAL Si <condición> Entonces <Instrucciones> Si no <Instrucciones> Fin si |  |
| HACER MIENTRAS Mientras <condición> Hacer <Instrucciones> Fin mientras |  |
| REPETIR HASTA Repetir <Instrucciones> Hasta que <condición> |  |
| CONDICIONAL MÚLTIPLE Según sea <variable> Hacer Caso opción 1: <Instrucciones> Caso opción 2: <Instrucciones> Caso opción 3: <Instrucciones> Otro caso: <Instrucciones> Fin según |  |

El grafo de flujo se crea a partir del diagrama de flujo con las equivalencias mostradas en estas imágenes.

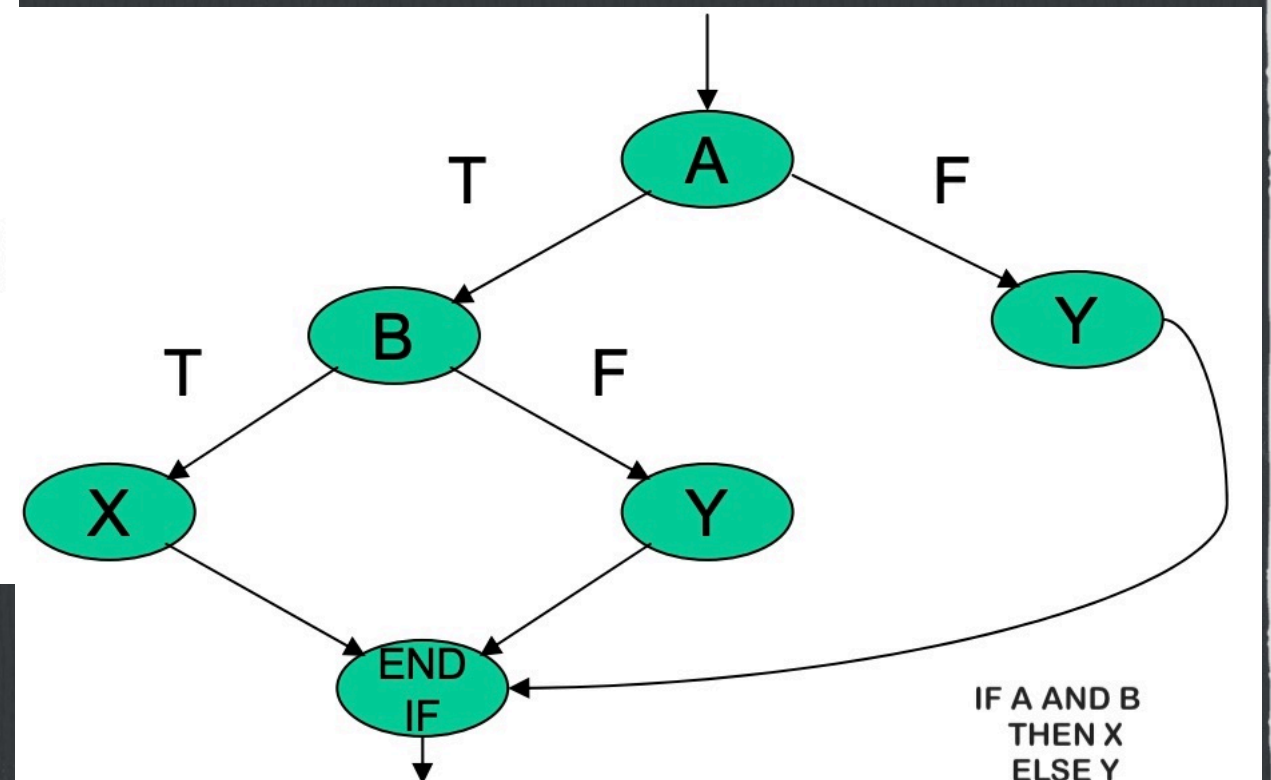
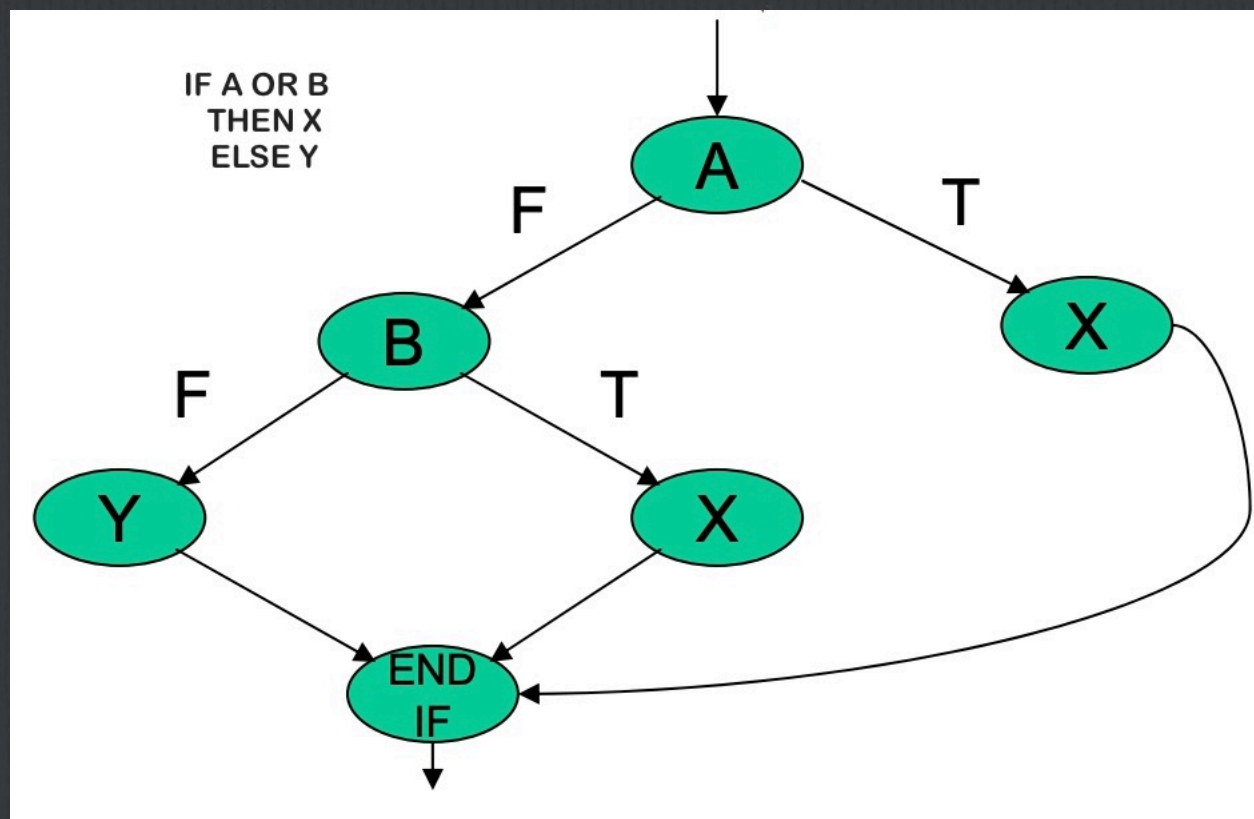
Las flechas del grafo de flujo se denominan aristas o enlaces. Cada arista termina en un nodo. Las aristas entre los nodos indican saltos en el flujo del programa.

Las áreas delimitadas por aristas y nodos se llaman regiones, el area exterior del grafo es otra región más.

El nodo que contiene una condición se llama nodo predicado y se caracteriza porque de él salen dos o más aristas.

PRUEBA DEL CAMINO BÁSICO: GRAFO DE FLUJO (III)

Las condiciones lógicas compuestas, se traducen en un grafo algo más complejo, pues se deben evaluar en base a una estructura lógica llamada “tabla de verdad”



PRUEBA DEL CAMINO BÁSICO: COMPLEJIDAD CICLOMÁTICA

- ☐ La complejidad ciclomática $V(G)$ es una métrica del software que proporciona una medida cuantitativa de la complejidad lógica de un programa. (wikipedia)
- ☐ Establece el número de caminos independientes del conjunto básico de caminos de ejecución de un programa.
- ☐ Dicho valor indica el número de casos de prueba que se deben ejecutar para asegurar que cada sentencia se ejecuta al menos una vez.
- ☐ La complejidad ciclomática puede calcularse de tres formas:
 - $V(G) = \text{Número de regiones del grafo.}$
 - $V(G) = \text{Aristas} - \text{Nodos} + 2.$
 - $V(G) = \text{Nodos predicado} + 1$

PRUEBA DEL CAMINO BÁSICO: CAMINOS INDEPENDIENTES

Un camino independiente es cualquier camino del programa que introduce, por lo menos, un nuevo conjunto de sentencias de proceso o una condición. En términos del diagrama de flujo, un camino independiente está constituido por lo menos por una arista que no haya sido recorrida anteriormente a la definición del camino.

Identificamos cada camino en base a los nodos que recorre de manera secuencial. Ex:

Camino 1: 1-10

Camino 2: 1-2-4-8-1-9

Camino 3: 1-2-3-5-7-8-1-9

Camino 4: 1-2-5-6-7-8-1-9

PRUEBA DEL CAMINO

BÁSICO: CASOS DE PRUEBA

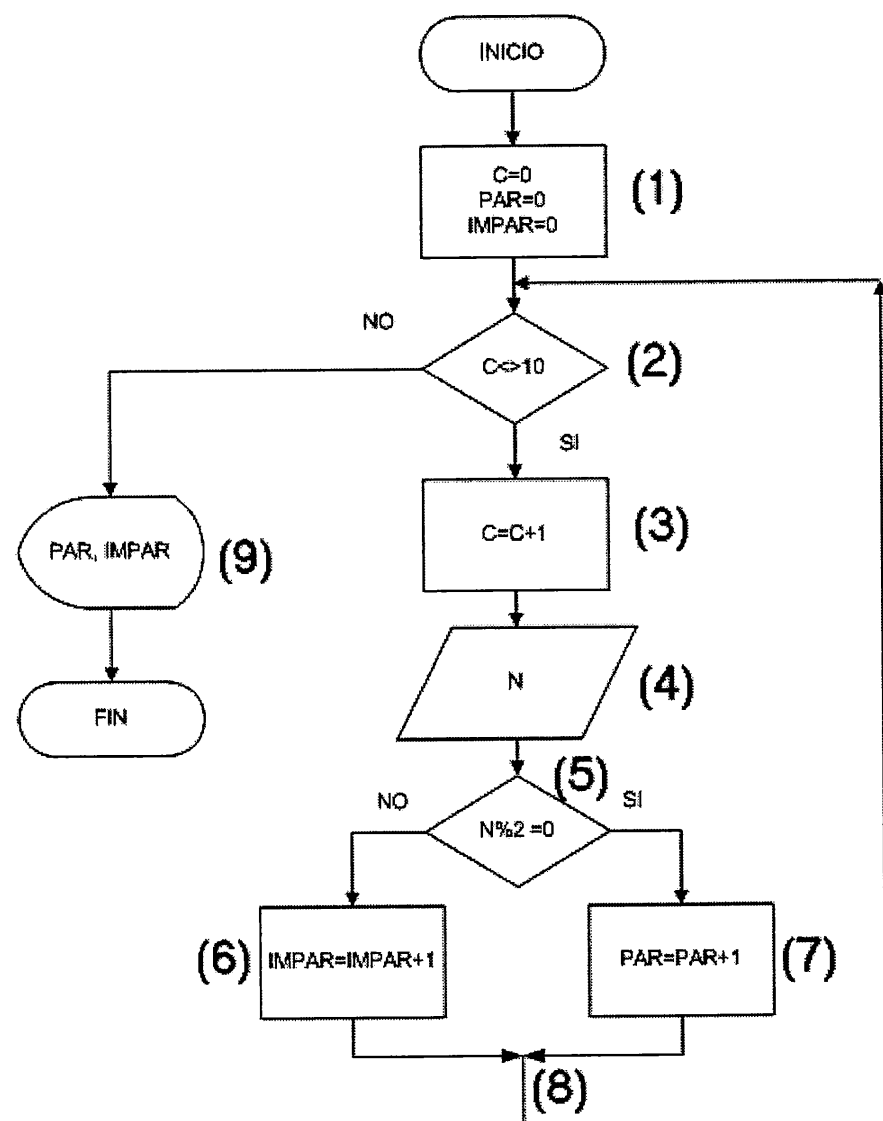
El último paso es la generación de datos para elaboración de los casos de prueba. Se deben escoger los datos de forma que las condiciones de los nodos predicados estén adecuadamente establecidas, con el fin de comprobar cada camino.

Tras confeccionar los casos de prueba, se ejecutan cada uno de estos y se comparan los resultados con los esperados. Una vez terminados todos los casos de prueba, podemos organizar la información en una tabla. Ex:

| Número del Camino | Caso de Prueba | Resultado Esperado |
|-------------------|----------------|--------------------|
| | | |
| | | |
| | | |

PRUEBA DEL CAMINO

BÁSICO: EJEMPLO (I)

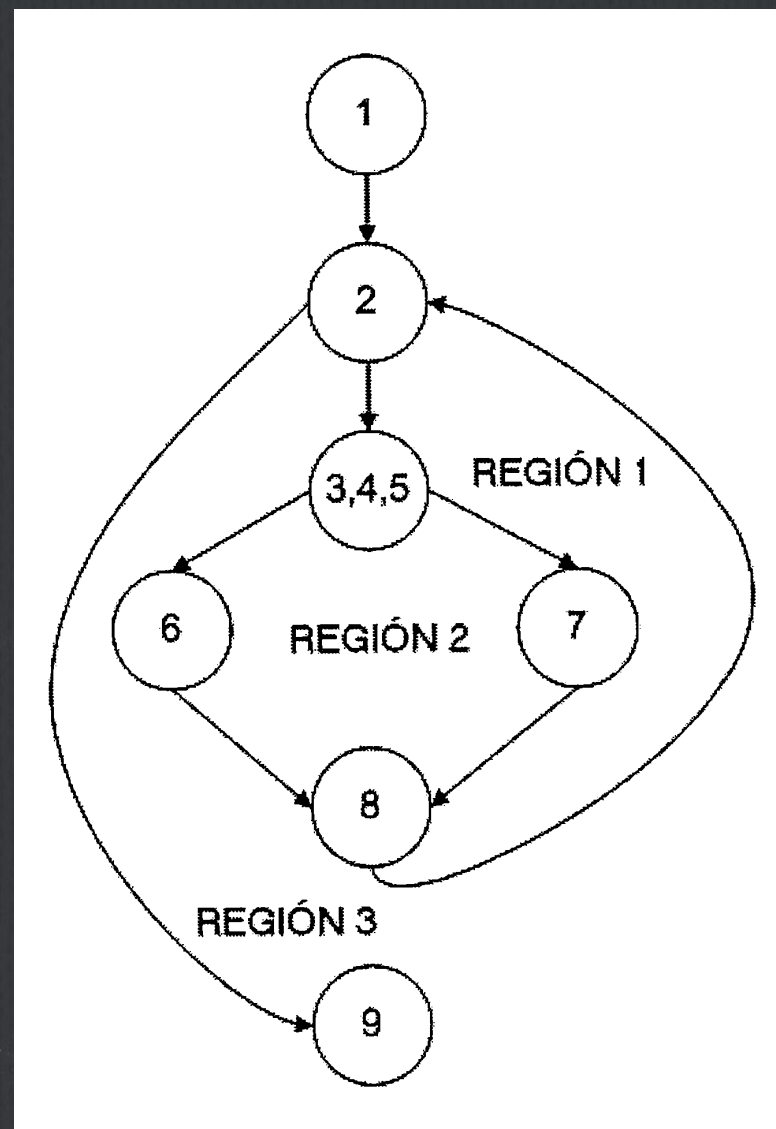


ENUNCIADO: diseña un programa que lee 10 números de teclado y muestra cuántos de los números leídos son pares y cuántos son impares.

En el paso 1 debemos realizar el diagrama de flujo y los posibles nodos (numerados) en él.

Aquí podemos identificar un bucle “while” y un condicional simple

PRUEBA DEL CAMINO BÁSICO: EJEMPLO (II)



PASO 1: El grafo generado es el mostrado en la imagen adjunta.

PASO 2: La complejidad del mismo es:

$$V(G) = N^{\circ} \text{ de regiones} = 3$$

$$V(G) = (A - N) + 2 = (8 - 7) + 2 = 3$$

$$V(G) = N_p + 1 = 2 + 1 = 3$$

PASO 3: Ahora debemos identificar los caminos:

Camino 1: 1 - 2 - 9

Camino 2: 1 - 2 - (3, 4, 5) - 6 - 8 - 2 - 9

Camino 3: 1 - 2 - (3,4,5) - 7 - 8 - 2 - 9

PRUEBA DEL CAMINO BÁSICO: EJEMPLO (II)

PASO 4: GENERACIÓN DE CASOS DE PRUEBA

| Camino | Caso de prueba | Resultado esperado |
|--------|---|---|
| 1 | Escoger algún valor de C tal que NO se cumpla la condición $C \neq 10$ $C = 10$ | Visualizar el número de pares y el de impares |
| 2 | Escoger algún valor de C tal que SÍ se cumpla la condición $C \neq 10$. Escoger algún valor de N tal que NO se cumpla la condición $N \% 2 = 0$ $C = 1, N = 5$ | Contar números impares |
| 3 | Escoger algún valor de C tal que SÍ se cumpla la condición $C \neq 10$. Escoger algún valor de N tal que SÍ se cumpla la condición $N \% 2 = 0$ $C = 2, N = 4$ | Contar números pares |