

Prueba de Caja Blanca

“Caso venta dulces”

Integrantes:
COLLAGUAZO PABLO
RODRIGUEZ JEICOL
VILAÑA ANTHONY

Fecha 16/1/2025

Prueba caja blanca de REGISTRO DE PRODUCTOS

CASO DE PRUEBA 1

1. CÓDIGO FUENTE

Ingreso del precio del producto

```
// Validar la entrada del nombre del producto
do {
    cout << "Ingrese el nombre del producto: ";
    cin.ignore(); // Limpiar el buffer de entrada
    getline(cin, nombreProducto);

    if (!esNombreValido(nombreProducto)) {
        cout << "Error: El nombre no puede estar vacío ni contener números." << endl;
    }
} while (!esNombreValido(nombreProducto)); // Repetir hasta que el nombre sea válido

p.nombre = nombreProducto;

// Ingresar y validar el precio del producto
p.precio = ingresarPrecio();

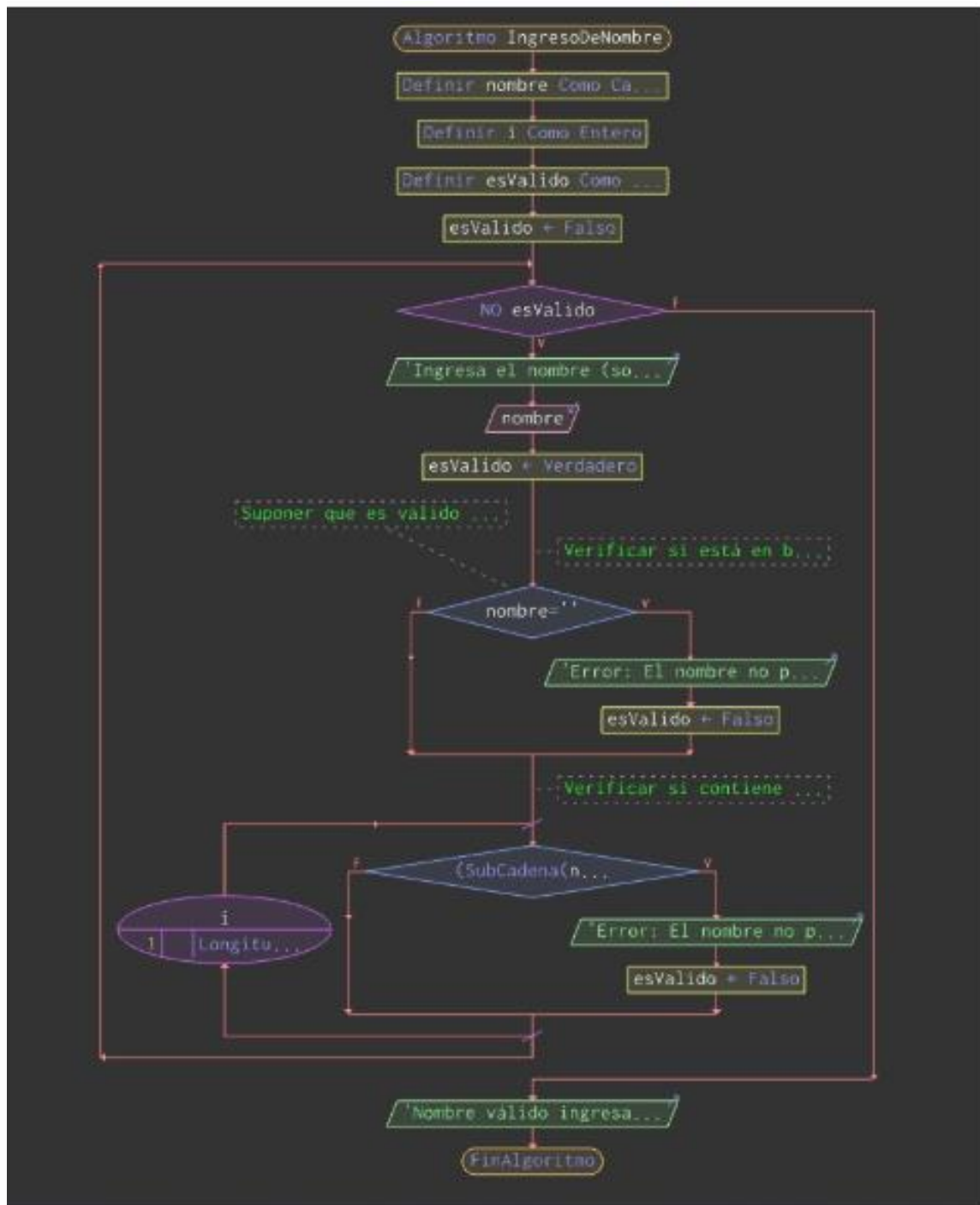
// Agregar el producto a la lista
productos.push_back(p);

// Sumar el precio del producto al valor total
valorTotal += p.precio;

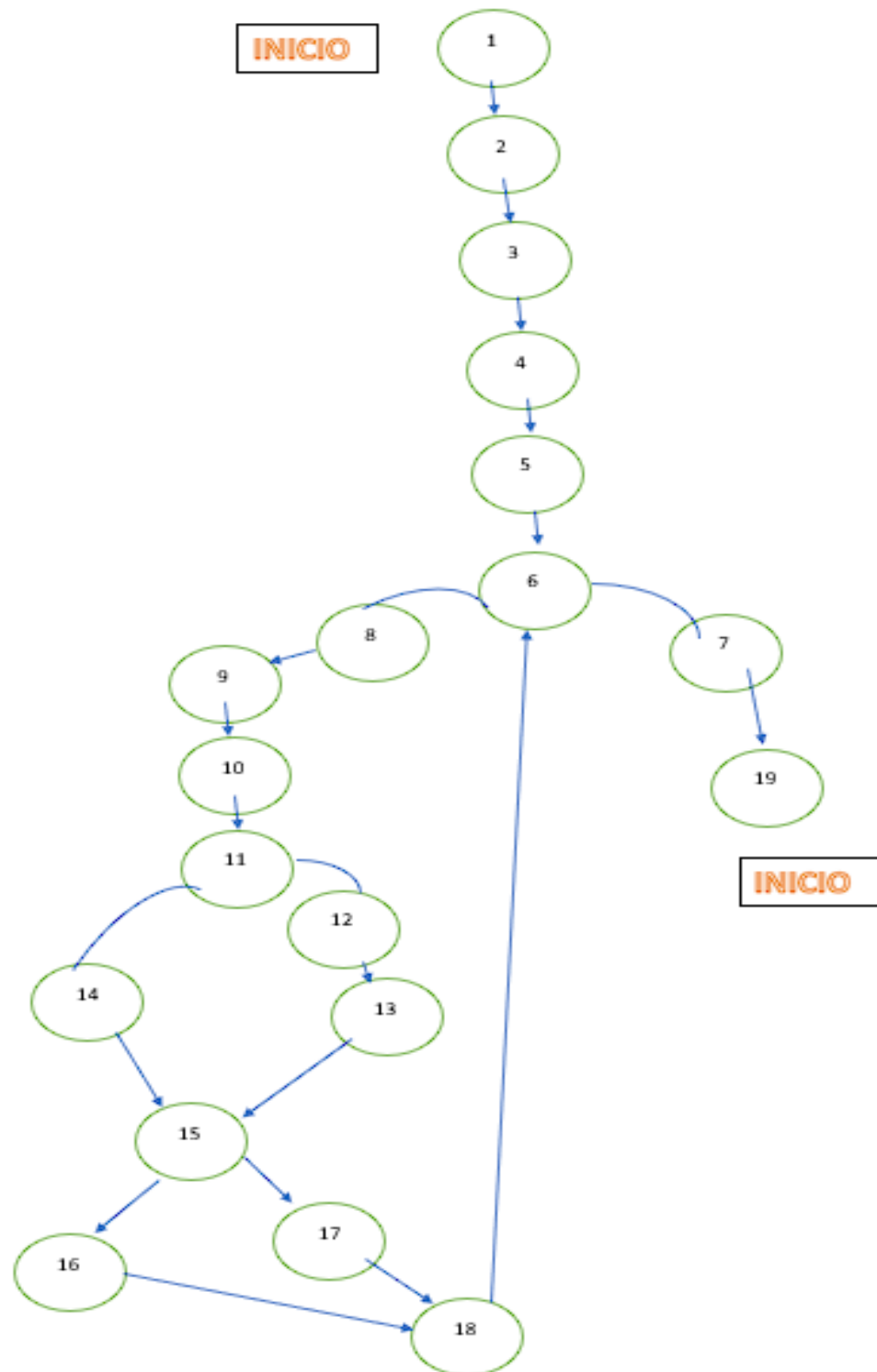
// Preguntar si desea agregar más productos
cout << "¿Desea ingresar otro producto? (s/n): ";
cin >> opcion;

} while (opcion == 's' || opcion == 'S');
```

2. DIAGRAMA DE FLUJO (DF)



3. GRAFO DE FLUJO (GF)



4. IDENTIFICACIÓN DE LAS RUTAS (Camino basico)

RUTAS

R1: 1-2-3-4-5-6-7-19

R2: 1-2-3-4-5-6-8-9-10-11-14-15-18-19

R3: 1-2-3-4-5-6-8-9-10-11-14-16-17-18-19

R4: 1-2-3-4-5-6-8-10-11-12-13-14-15-18-19

5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- $V(G) = \text{número de nodos predcados(decisiones)} + 1$
 $V(G) = 3 + 1 = 4$
- $V(G) = A - N + 2$
 $V(G) = 21 - 19 + 2 = 4$

DONDE:

P: Número de nodos predcado

A: Número de aristas

N: Número de nodos

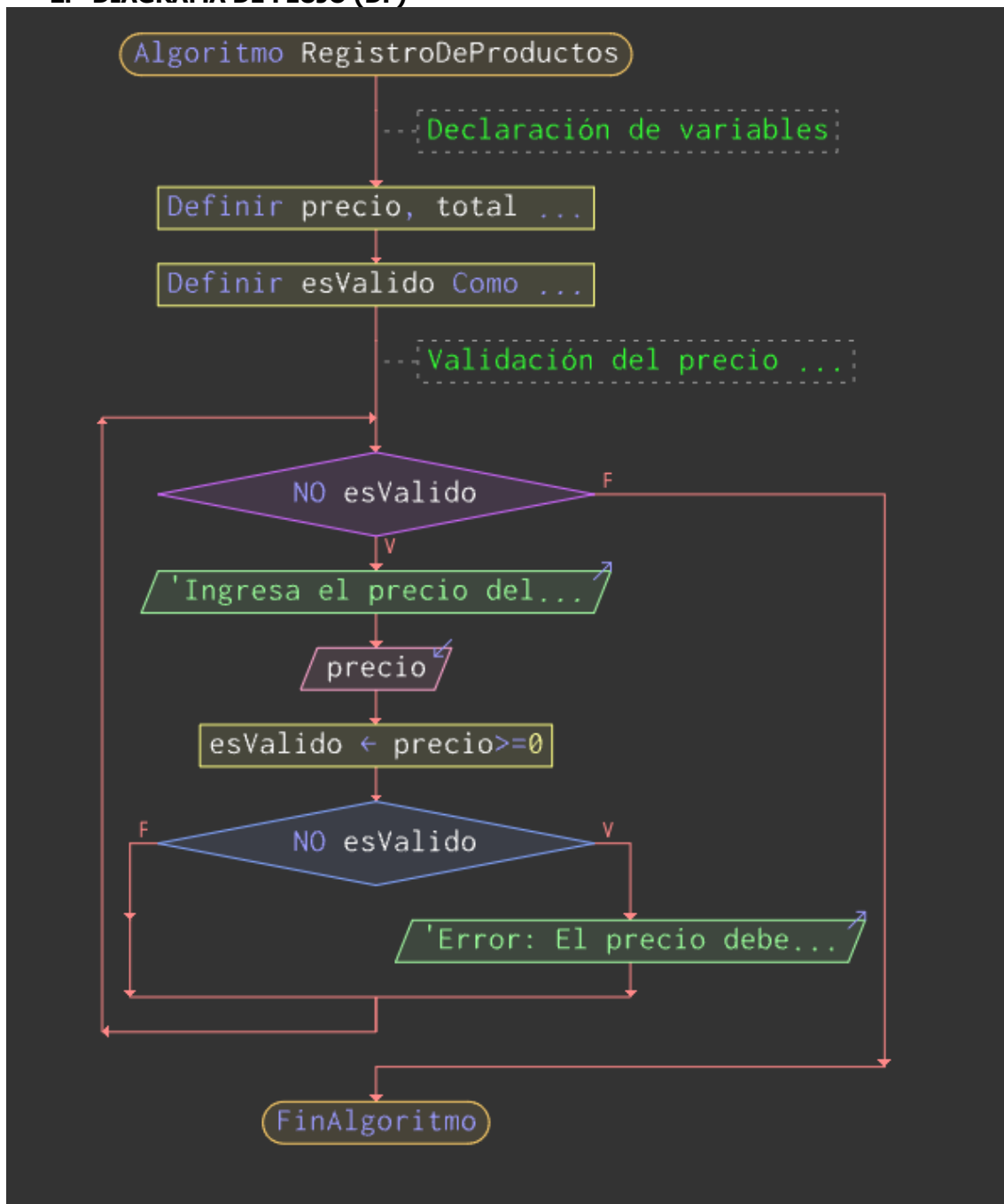
Prueba caja blanca de INGRESO DE PRECIOS

1. CÓDIGO FUENTE PRECIOS

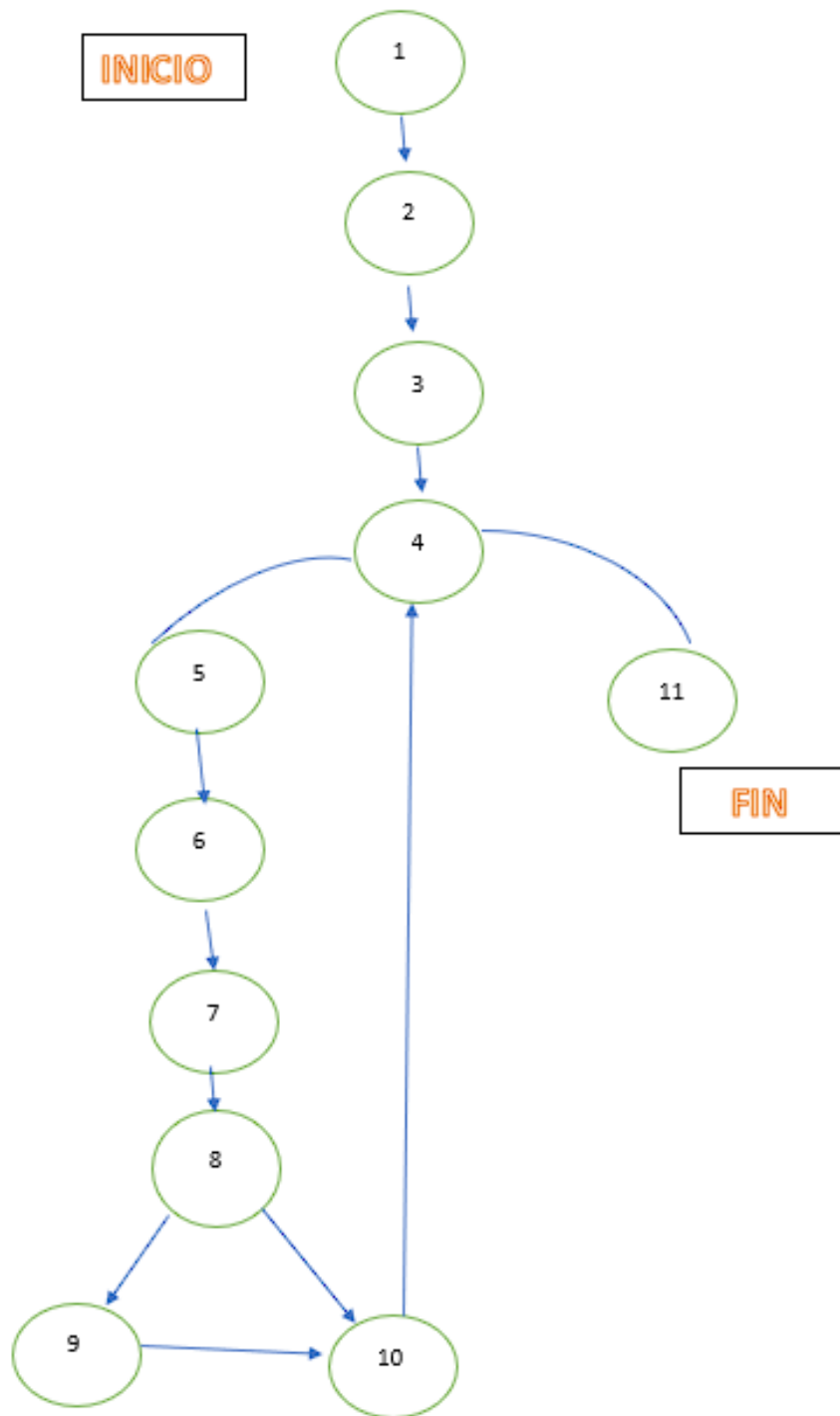
```
// Función para validar el precio (número positivo y no letras)
float ingresarPrecio() {
    float precio;
    while (true) {
        cout << "Ingrese el precio del producto: ";
        cin >> precio;

        if (cin.fail() || precio < 0) { // Si la entrada falla (no es un número) o es negativa
            cout << "Error: ingrese un valor numérico positivo." << endl;
            cin.clear(); // Limpiar el estado de error de cin
            cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Descartar la entrada incorrecta
        } else {
            break; // Si el precio es válido, salir del bucle
        }
    }
    return precio;
}
```

2. DIAGRAMA DE FLUJO (DF)



3. GRAFO DE FLUJO (GF)



4. IDENTIFICACIÓN DE LAS RUTAS (Camino básico)

Determinar en base al GF del numeral 4

RUTAS

R1: 1-2-3-4-11

R2: 1-2-3-4-5-6-7-8-9-10-11

R3: 1-2-3-4-5-6-7-8-10-11

5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- $V(G) = \text{número de nodos predichos (decisiones)} + 1$
 $V(G) = 2 + 1 = 3$
- $V(G) = A - N + 2$
 $V(G) = 12 - 11 + 2 = 3$

DONDE:

P: Número de nodos predichos

A: Número de aristas

N: Número de nodos