

# Prueba de Caja Blanca

---

## ***“SISTEMA DE INVENTARIO DE FRUTOS SECOS”***

### **Integrantes:**

- Alvear Alexnader
- Mateo Velecela
- Antony Campoverde

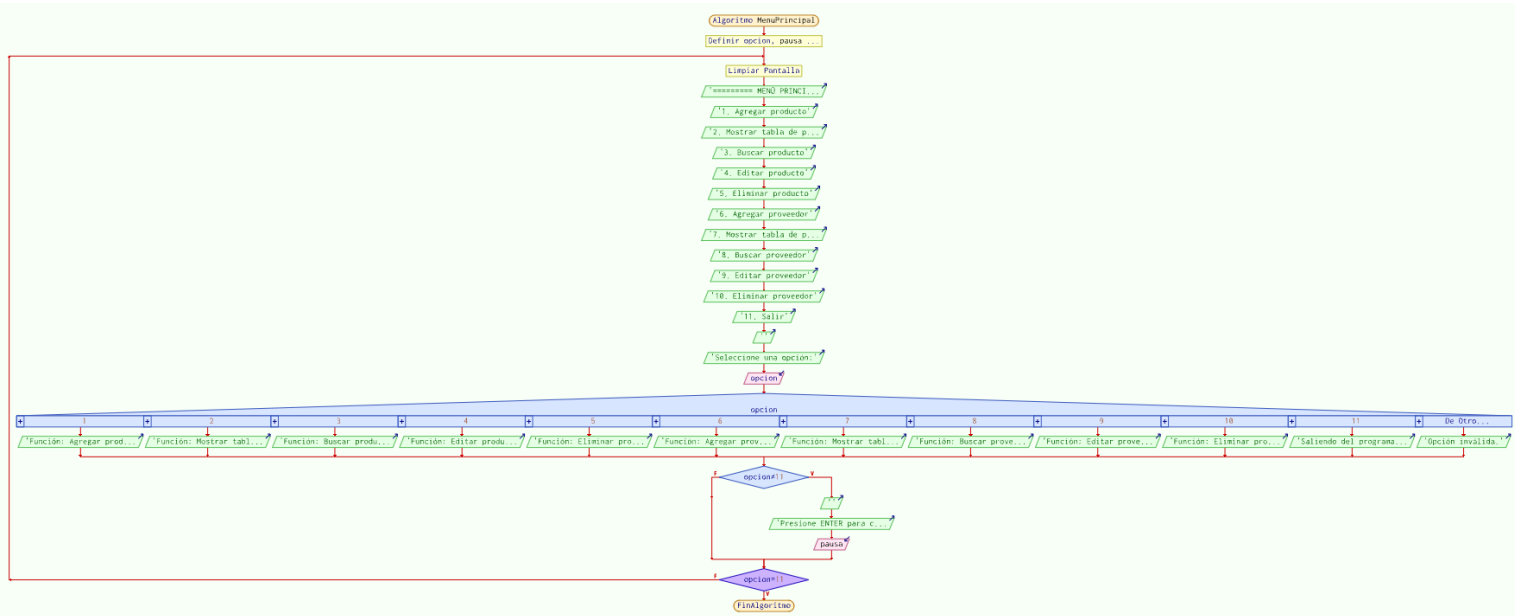
**Fecha: 2025/07/24**

## Prueba caja blanca : Rellar id con ceros

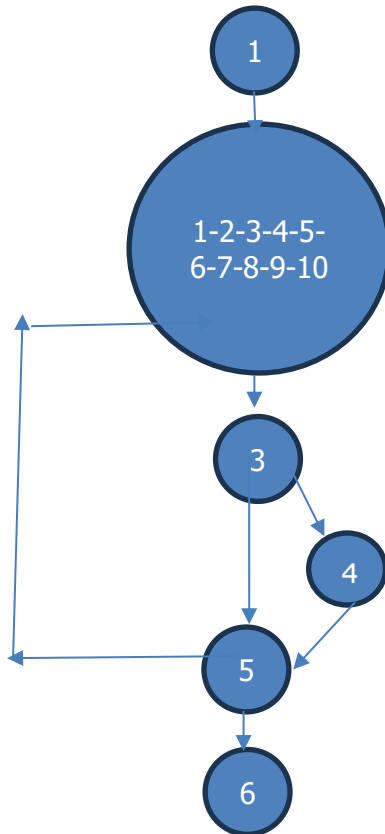
### 1. CÓDIGO FUENTE

```
669         do {
670             system(CLEAR);
671             imprimirMenu();
672             if (scanf("%d", &opcion) != 1) {
673                 printf("Entrada invalida.\n");
674                 limpiarBuffer();
675                 opcion = 0;
676                 continue;
677             }
678             limpiarBuffer();
679
680             switch (opcion) {
681                 case 1: agregarProducto(); break;
682                 case 2: mostrarTablaProductos(); break;
683                 case 3: buscarProducto(); break;
684                 case 4: editarProducto(); break;
685                 case 5: eliminarProducto(); break;
686                 case 6: agregarProveedor(); break;
687                 case 7: mostrarTablaProveedores(); break;
688                 case 8: buscarProveedor(); break;
689                 case 9: editarProveedor(); break;
690                 case 10: eliminarProveedor(); break;
691                 case 11: printf("Saliendo...\n"); break;
692                 default: printf("Opcion invalida.\n"); pausa(); break;
693             }
694         } while (opcion != 11);
695
696         return 0;
697     }
```

### 2. DIAGRAMA DE FLUJO (DF) PSEINT



### 3. GRAFO DE FLUJO (GF)



### 4. IDENTIFICACIÓN DE LAS RUTAS (Camino básico)

#### RUTAS

**R1: 1-2-3-5-6**

**R2: 1-2-3-4-5-2**

**R3:**

### 5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- $V(G) = \text{número de nodos predicados(decisiones)} + 1$   
 $V(G) = 2 + 1 = 3$
- $V(G) = A - N + 2$   
 $V(G) = 7 - 6 + 2 = 3$

DONDE:

**P:** Número de nodos predicado

**A:** Número de aristas

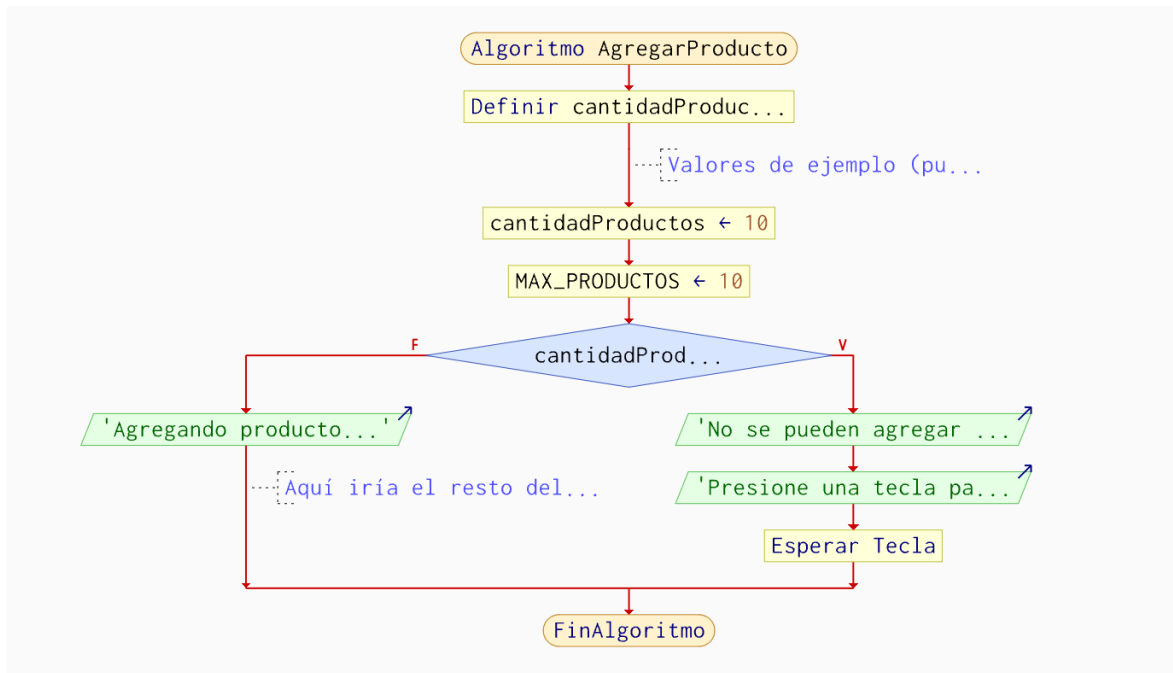
**N:** Número de nodos

## Prueba caja blanca de Requisito N° 2: Como agregar productos

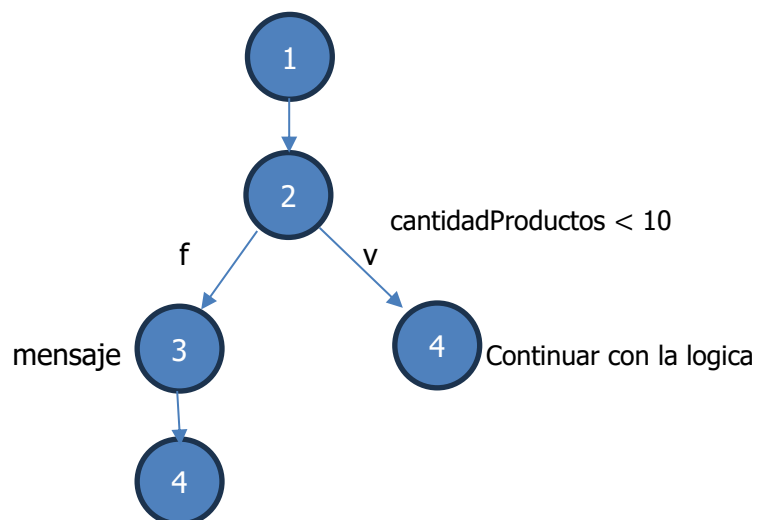
### 1. CÓDIGO FUENTE

```
150 void agregarProducto() {  
151     if (cantidadProductos >= MAX_PRODUCTOS) { // MAX_PRODUCTOS 100  
152         printf("No se pueden agregar mas productos.\n");  
153         pausa();  
154         return;  
155     }
```

### 2. Diagrama de flujo



### 3. Grafo de flujo (GF)



#### 4. IDENTIFICACIÓN DE LAS RUTAS (Camino básico)

##### RUTAS

R1: 1-2-3-4

R2: 1-2-4

#### 5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- a.  $V(G) = \text{número de nodos predados(decisiones)} + 1$   
 $V(G) = 1 + 1 = 2$
- b.  $V(G) = A - N + 2$   
 $V(G) = 4 - 4 + 2 = 2$

DONDE:

**P:** Número de nodos predado

**A:** Número de aristas

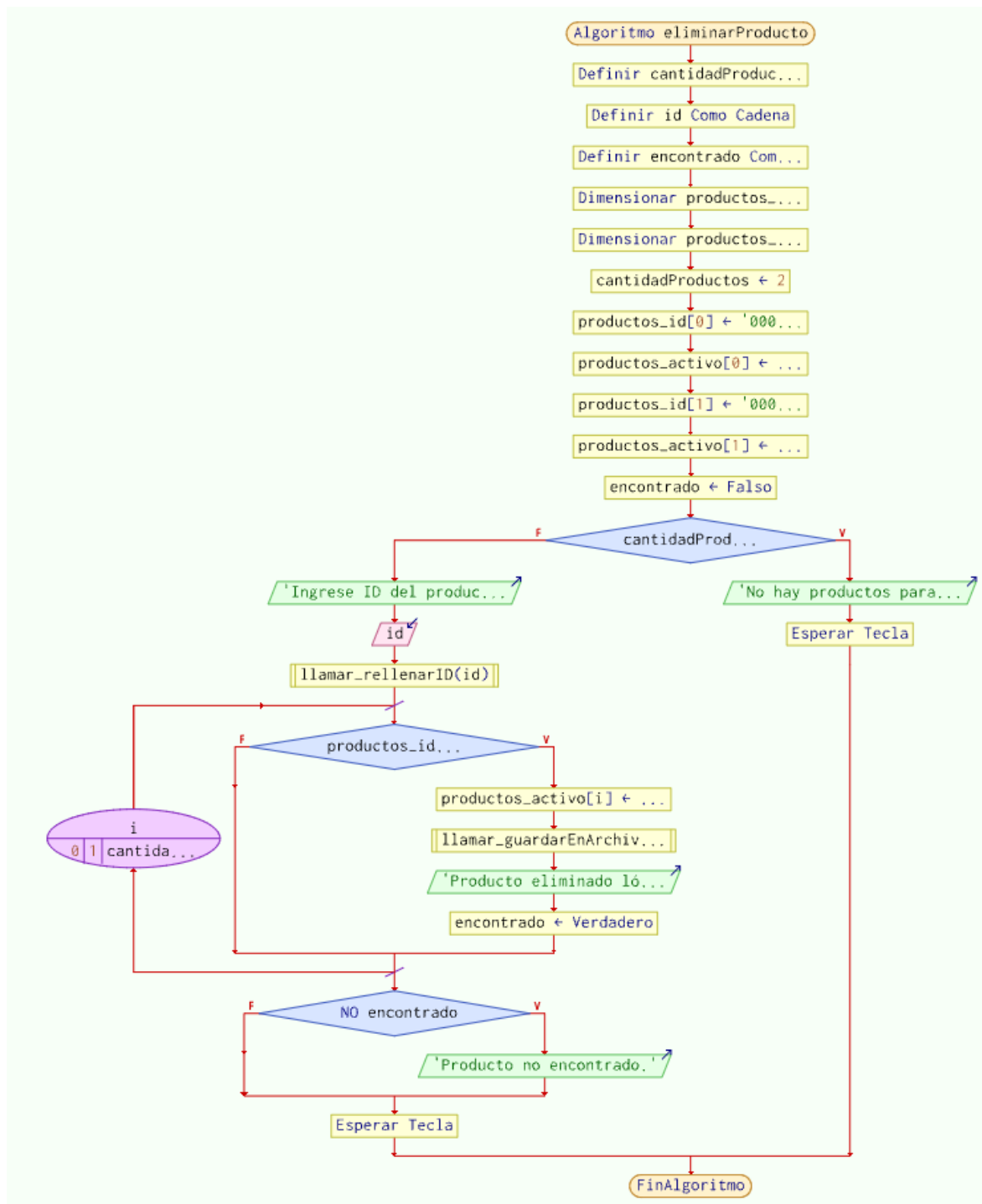
**N:** Número de nodos

**Prueba caja blanca de Requisito N° 3:** Eliminar producto

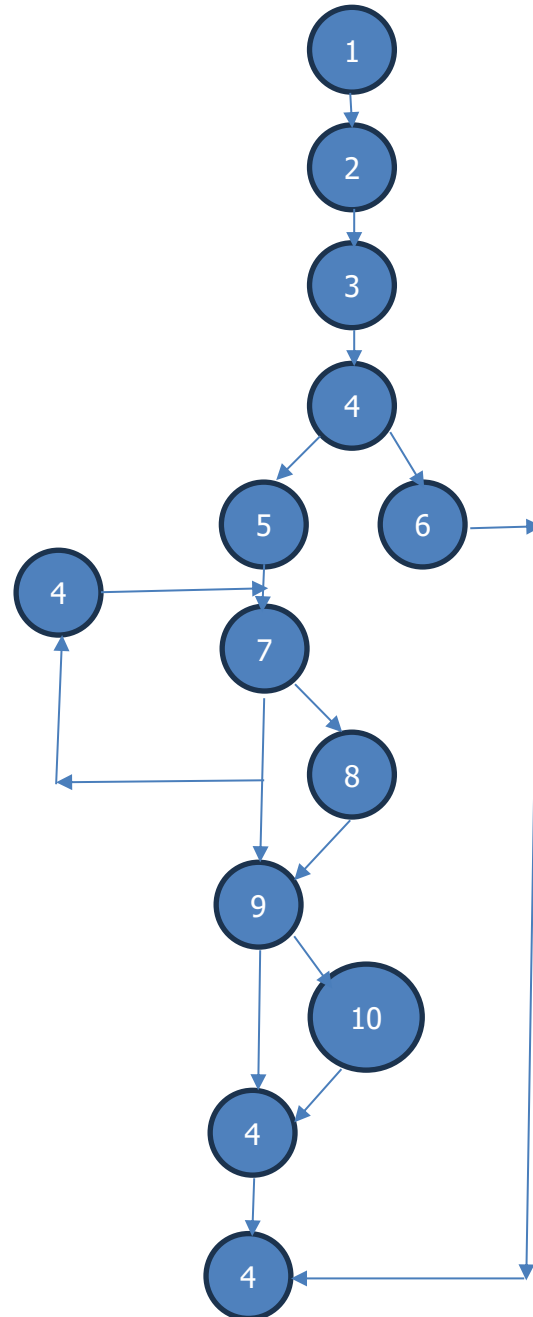
##### 1. CÓDIGO FUENTE

```
360
361 void eliminarProducto() {
362     if (cantidadProductos == 0) {
363         printf("No hay productos para eliminar.\n");
364         pausa();
365         return;
366     }
367     char id[20];
368     printf("Ingrese ID del producto a eliminar (numeros, se rellenara a 10 digitos): ");
369     fgets(id, sizeof(id), stdin);
370     id[strcspn(id, "\n")] = '\0';
371     rellenarID(id);
372
373     for (int i = 0; i < cantidadProductos; i++) {
374         if (strcmp(productos[i].id, id) == 0 && productos[i].activo) {
375             productos[i].activo = 0;
376             guardarEnArchivos();
377             printf("Producto eliminado lógicamente.\n");
378             pausa();
379             return;
380         }
381     }
382     printf("Producto no encontrado.\n");
383     pausa();
384 }
```

## 2. DIAGRAMA DE FLUJO (DF) PSEINT



### 3. GRAFO DE FLUJO (GF)



### 4. IDENTIFICACIÓN DE LAS RUTAS (Camino básico)

#### RUTAS

R1:

R2:

### 5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- $V(G) = \text{número de nodos predicaos(decisiones)} + 1$   
 $V(G) = 3 + 1 = 4$

- $V(G) = A - N + 2$   
 $V(G) =$

DONDE:

**P:** Número de nodos predado

**A:** Número de aristas

**N:** Número de nodos