PARCIAL 1

RESUMEN PPSS

Curso 2020-2021

Francisco Javier Pérez Martínez

Contenido

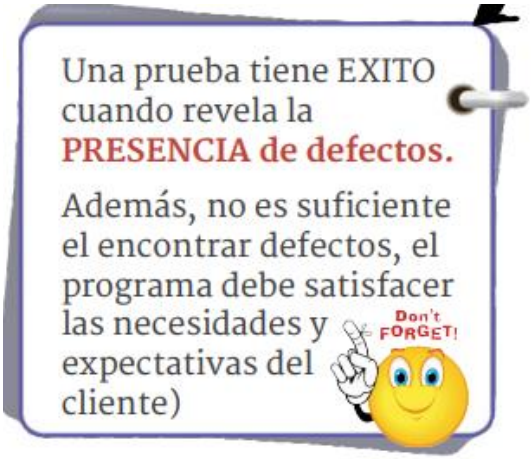
TEMA 1: Caja Blanca

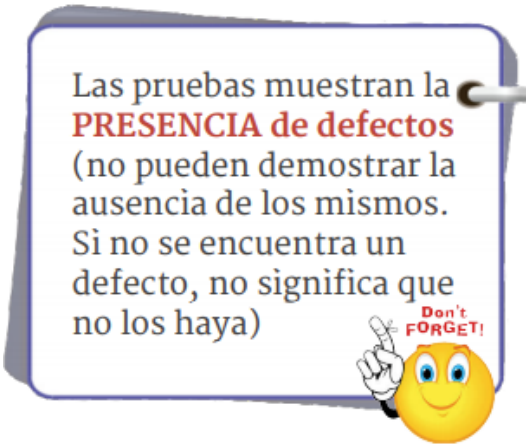
- Objetivo: obtener una tabla de casos de prueba a partir del conjunto de comportamientos programados.

- El conjunto obtenido debe detectar el máximo número posible de defectos en el código, con el mínimo número posible de "filas".

Control flow testing: Método del camino básico

- Paso 1: Análisis de código: Construcción del CFG.
- Paso 2: Selección de caminos: Cálculo de CC y caminos independientes.
- Paso 3. Obtención del conjunto de casos de prueba.



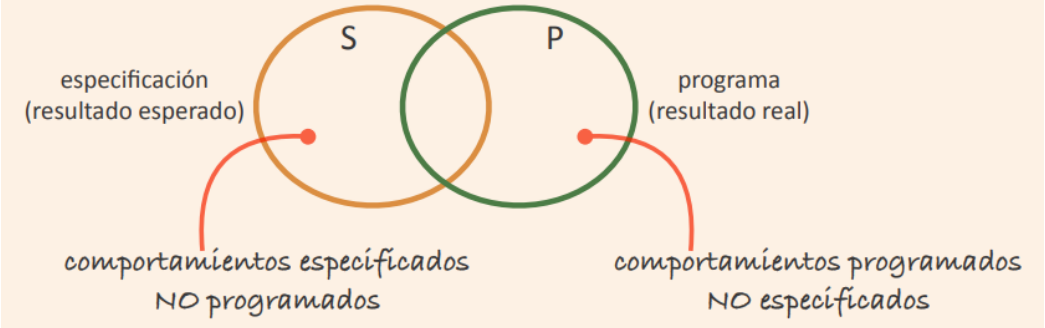


1.1 Actividades del proceso de pruebas

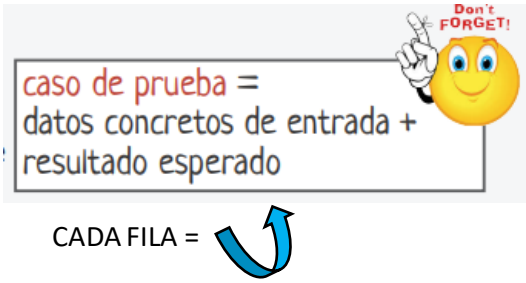
1. Planificación y control de las pruebas
2. Diseño de las pruebas
3. Implementación y ejecución de las pruebas
4. Evaluación del proceso de pruebas y emitir un informe

1.2 Pruebas y comportamiento

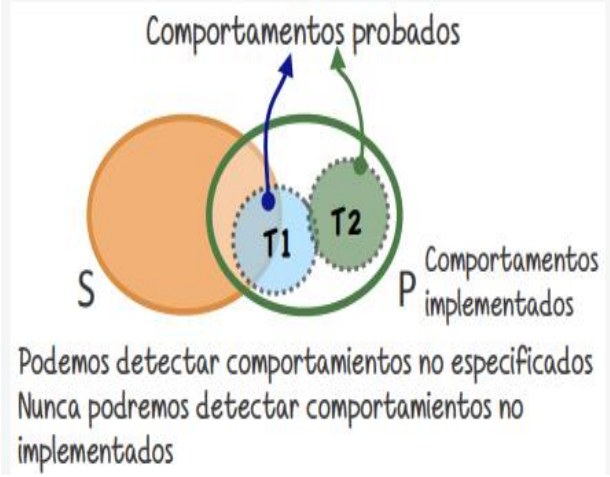
S y P deberían ser idénticos!!!



El objetivo principal al realizar un conjunto de casos de prueba mediante un método de diseño de pruebas es encontrar el máximo número posible de defectos con el mínimo número de casos de prueba.



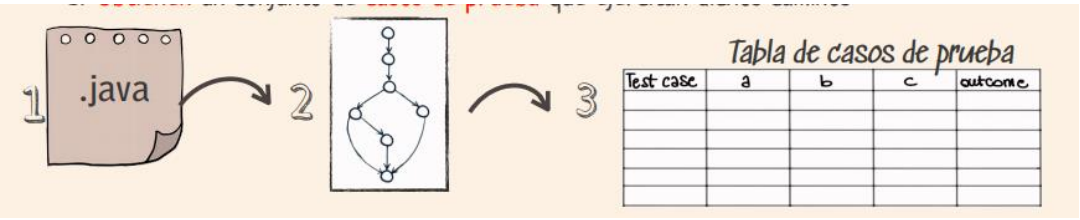
1.3 Diseño de casos de prueba (CAJA BLANCA)



- Consiste en determinar los valores de entrada de los casos de prueba a partir de la IMPLEMENTACIÓN.
- El resultado esperado se obtiene SIEMPRE de la especificación.
- Los comportamientos probados podrán estar o no especificados.
- Es esencial conocer conceptos de teoría de grafos para entender bien esta aproximación.

1.4 Principios para cualquier método basado en el código

1. Analizan el código y obtienen una representación en forma de GRAFO.
2. Seleccionan un conjunto de caminos en el grafo según algún criterio.
3. Obtienen un conjunto de casos de prueba que ejercitan dichos caminos.



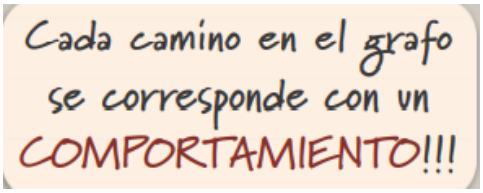
Observaciones:

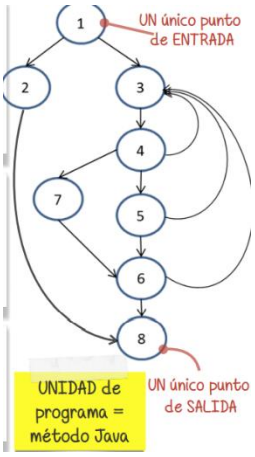
- Dependiendo del método utilizado se obtendrán DIFERENTES conjuntos de casos de prueba. ¡¡¡Pero éste debe ser EFECTIVO y EFICIENTE!!!
- Técnicas o métodos estructurales son costos de aplicar, sólo se usan a nivel de UNIDADES de programa.
- Los métodos estructurales NO pueden DETECTAR TODOS los defectos en el programa.

1.5 Grafo de flujo de control (CFG)

Uso de GRAFO DIRIGIDO.

- Cada **nodo** representa una o más sentencias secuenciales y/o una ÚNICA CONDICIÓN. (único punto de entrada y único punto de salida).
  - Cada nodo etiquetado con un entero cuyo valor será único.
  - Anotar a la derecha de un nodo condición.
- Las **aristas** representan el flujo de ejecución entre dos conjuntos de sentencias.
  - Si un nodo contiene una condición etiquetaremos las aristas con T o F.





1.6 Método del camino básico

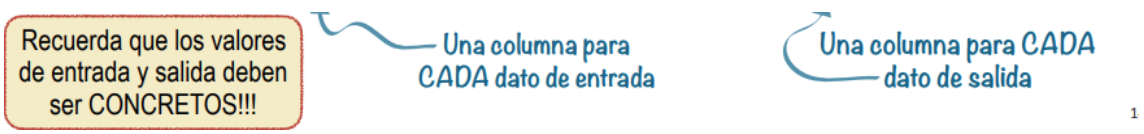
El objetivo de este método es ejecutar TODAS las sentencias del programa, al menos una vez para garantizar así que TODAS las condiciones se ejecutan ya sea V/F.

- El Nº de caminos independientes (CI) determinará el Nº de filas de la tabla.

Pasos:

1. Construir el CFG a partir del código a probar.
2. Calcular la complejidad ciclomática (CC).
3. Obtener los CI del grafo.
4. Determinar los datos de entrada (y salida esperada) de la unidad a probar, ejercitando así todos los CI.

NOTA: el resultado esperado se obtendrá de la ESPECIFICACIÓN de la unidad a probar.



1

1.6.1 Formas de calcular CC

- CC = nº de arcos – nº de nodos + 2.
- CC = nº de regiones
- CC = nº de condiciones + 1

¡CUIDADO!: Las dos últimas formas de cálculo son aplicables SOLO si el código es totalmente estructurado (no saltos incondicionales)

El valor de CC indica el MÁXIMO número de CI en el grafo.

1.6.2 Caminos independientes

Buscar como máximo tantos CI como valor obtenido de CC.

- Cada CI contiene un nodo, o bien una arista, que no aparece en los caminos independientes anteriores.
- Con ellos recorremos TODOS los nodos y TODAS las aristas del grafo.

Es posible que con un nº inferior al CC recorramos todos los nodos y aristas.