



**DOCUMENTOS RELACIONADOS CON EL DISEÑO
DE LAS PRUEBAS SEGUN EL ESTANDAR IEEE std. 829**

**APLICACIÓN AL SOFTWARE
JUEGO DE SCELULAS**

ESTÁNDAR IEEE 829

PLAN DE PRUEBAS

1. Identificador único del documento (para la gestión de configuración).

DOC-003

2. Introducción y resumen de elementos y características a probar.

El documento a continuación describe el alcance, la aproximación, los recursos y la planificación y las actividades necesarias. Identifica elementos de prueba, las características que deben probarse, las tareas de prueba, lo que hará cada tarea.

Los elementos a ser probados son:

- Software
- Documentación

3. Elementos

3.1 software que se van a probar (por ejemplo, programas o módulos).

MODULO ENTRADA

sub. Modulo -Definición de Variables:

Define las variables de la matriz las cuales son fila, col, FILAS, COLS, etc.

sub. Modulo -Inicializa matriz y pone las células iniciales:

Se introducen los datos por defecto aleatoria mente

MODULO PROCESO

sub. Modulo -Imprime en pantalla la matriz de la población Elige un vecino aleatoria mente:

Se inicializa en la pantalla las celulas blancas y negras aleatoriamente y luego toma su vecino aleatorio y toma el valor para poder realizar el cambio de color.

sub. Modulo -Explora la matriz y averigua que habitante hay:

Registra la cantidad de habitantes e imprime en la pantalla (en un extremo) con el numero exacto de celulas blancas y negras.

MODULO SALIDA

sub. Modulo -Visualización en la matriz

3.2 Documentos a probar

- Doc Diseño
- Doc Analisis

4.Características que se van a probar.

1. fluidez de datos
2. independencia de módulos
3. soporte del software
4. interfaz de usuario

5.Características que no se prueba.

- Errores relacionados con el tiempo.
- Condiciones de error no detectadas.
- Condiciones especiales de los datos.
- Invalidez de la información mostrada por pantalla.
- Interacción con tareas en background.
- Fallos de configuración/compatibilidad con software
- Incapacidad de soportar el volumen de carga o fallos hard.

6.Enfoque general de la prueba(actividades, técnicas, herramientas, etc).

PRUEBA DEL DISEÑO

- Las pruebas del diseño van encaminadas a asegurar que la arquitectura propuesta es coherente, consistente y completa.

PRUEBAS DE UNIDAD

- Pretenden probar que los fragmentos individuales (unidades) que forman el sistema cumplen las especificaciones y tienen el comportamiento esperado.

PRUEBAS DE INTEGRACIÓN

- Se prueban las funcionalidades, rendimiento, fiabilidad, etc. del sistema, sus relaciones con el exterior, etc.

PRUEBAS DE REQUISITOS

- Diferentes técnicas de captura y análisis de requisitos (prototipos, casos de uso, etc.)

- El resultado es una descripción de las funciones del sistema

PRUEBA DE INTERFAZ

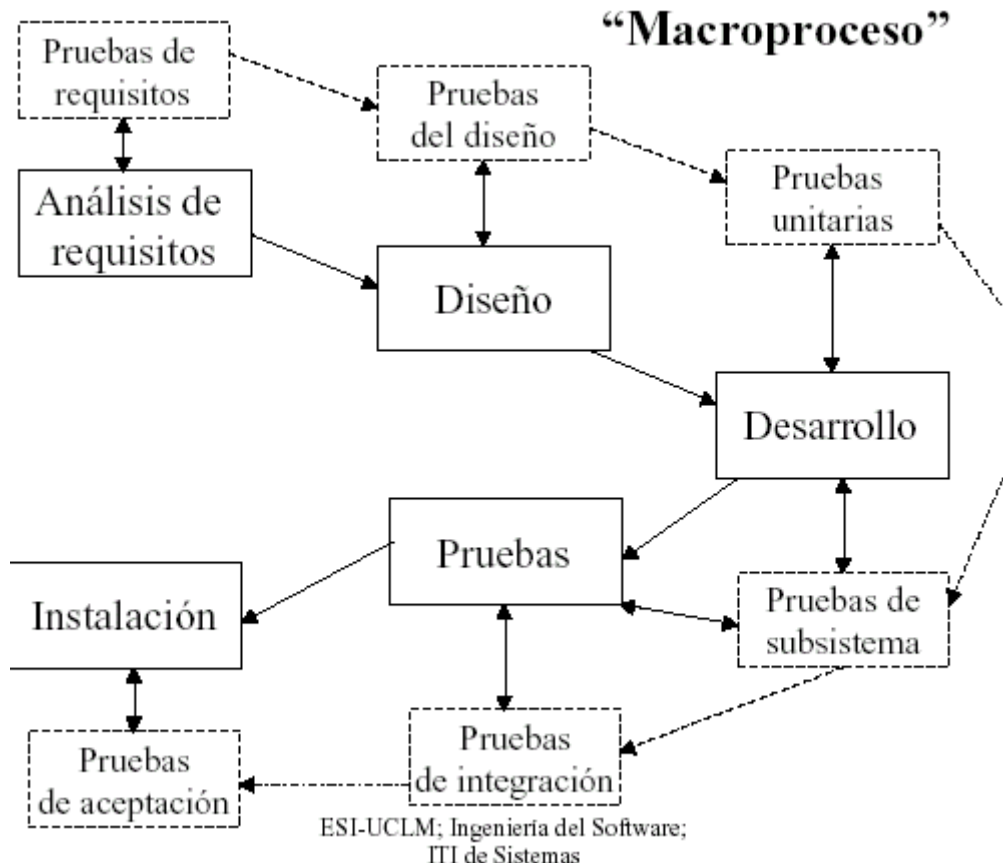
Parece que ya hemos logrado proporcionar a nuestros usuarios una interfaz gráfica bien organizada, similar a la de otras aplicaciones, utilizable con el teclado, con ayudas en toda la interfaz y en su idioma.

PRUEBA DE GRAFOS

Un criterio más riguroso se basa en la completitud ya no aplicado a las sentencias sino a los arcos del grafo de flujo de control del programa.

Nuevamente, asumiremos un lenguaje estructurado a bloques para nuestro análisis.

7. Criterios de paso/fallo para cada elemento.



8. Criterios de suspensión y requisitos de reanudación.

No existe

9. Documentos a entrega (como mínimo, los descritos en el estándar).

- Informe de Grafos
- Informe de Resistencia
- Informe de unidad
- Informe de integración
- Informe de interfaz grafica
- Informe de documentación y ayuda

10.Actividades de preparación y ejecución de pruebas.

Organización de Equipos

Jefe de equipo

SHIRLEY VELIZ BUSTILLOS

- Preparación de casos de pruebas
- Ejecución de pruebas
- Datos de la prueba
- Preparar informe

11.Necesidades de entorno.

En cuanto a:

SOFTWARE y HADWARE:

Sistema operativo MS-dos o Windows

Un computador con requerimiento mínimo de un microprocesador 486

DOCUMENTACION:

Absoluta comodidad, tranquilidad.

12.Responsabilidades en la organización y realización de las pruebas.

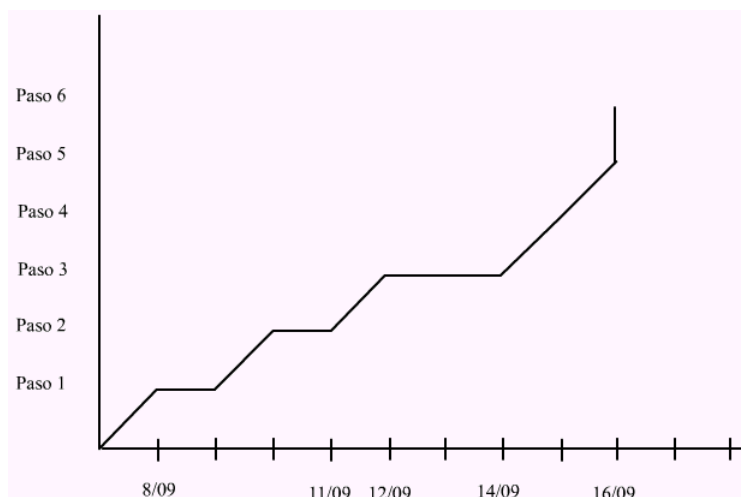
Pruebas de Documentación: Shirley Veliz Bustillos

Pruebas de software: Jorge Carlos Mendieta A.

13.Necesidades de personal y de formación.

Que sepa la utilización de sistemas operativos (MS-dos) y lenguaje de programación (C++)

14.Esquema de tiempos (con tiempos estimados, hitos,etc.)



15. Riesgos asumidos por el plan y planes de contingencias para cada riesgo.

Copias de seguridad
Frecuencia
Periodicidad
Plan de contingencias
Prever fallos críticos
Procedimientos alternativos
Tratamiento de errores
Posibilidad de error recuperación
Planificación contenido mensajes de error

16. Aprobaciones y firmas con nombre y puesto desempeñado.

Jorge Carlos Mendieta

Shirley Veliz Bustillos

ESPECIFICACIÓN DEL DISEÑO DE PRUEBAS

1. **Identificador (único) para la especificación.** Proporcionar también una referencia del plan asociado (si existe).

No existe

2. **Características a probar de los elementos software (y combinaciones de características).**

- fluidez de datos
- independencia de módulos
- soporte del software
- interfaz de usuario

3. **Detalles sobre el plan de pruebas del que surge este diseño, incluyendo las técnicas de prueba específica y los métodos de análisis de resultados.**

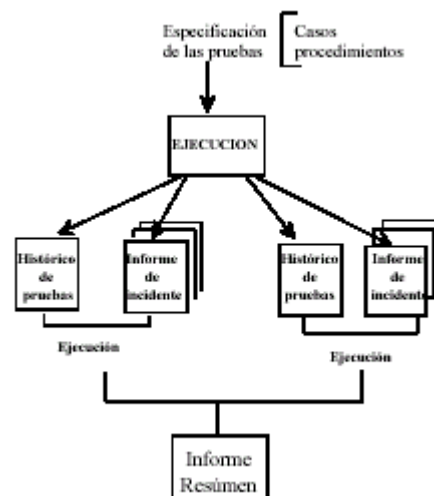
4. **Identificación de cada prueba:**

*Identificador.

*Casos que se van a utilizar.

*Procedimientos que se van a seguir.

5. **Criterios de paso/fallo de la prueba** (criterios para determinar si una característica o combinación de características ha pasado con éxito la prueba o no).



HISTORICO DE PRUEBAS

El histórico de pruebas (test log) documenta todos los hechos relevantes ocurridos durante la ejecución de las pruebas

HISTORICO DE PRUEBAS

Identificador

Descripción de la prueba: elementos probados y entorno de la prueba
Anotación de datos sobre cada hecho ocurrido (incluido el comienzo y el final de la prueba)

- Fecha y hora
 - Identificador de informe de incidente
- Otras informaciones

INFORME DE INCIDENTE

El informe de incidente (test incident report) documenta cada incidente (por ejemplo, una interrupción en las pruebas debido a un corte de electricidad, bloqueo del teclado, etc.) ocurrido en la prueba y que requiera una posterior investigación.

INFORME DE INCIDENTE

Identificador
Resumen del incidente
Descripción de datos objetivos (fecha/hora, entradas, resultados esperados, etc)
Impacto que tendrá sobre las pruebas

INFORME RESUMEN DE PRUEBAS

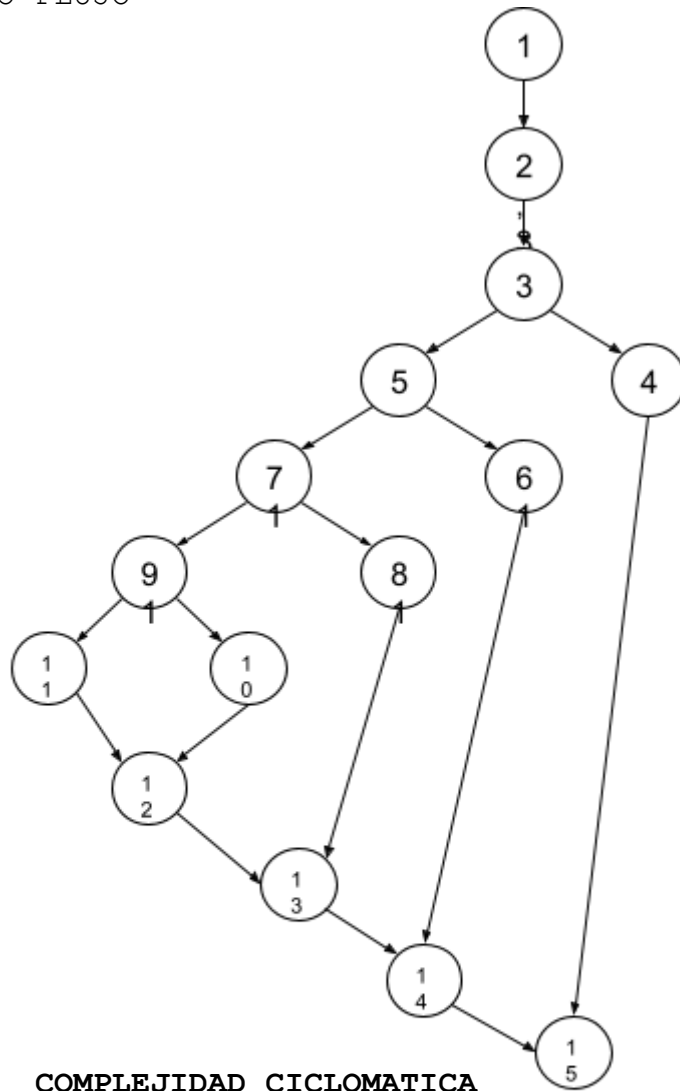
El informe resumen (test summary report) resume los resultados de las actividades de prueba (las señaladas en el propio informe) y aporta una evaluación del software basada en dichos resultados

INFORME RESUMEN DE LAS PRUEBAS

Identificador
Resumen de la evaluación de los elementos probados
Variaciones del software respecto a su especificación de diseño, así como las variaciones en las pruebas
Valoración de la extensión de la prueba (cobertura lógica, funcional, de requisitos, etc.)
Resumen de los resultados obtenidos en las pruebas
Evaluación de cada elemento software sometido a prueba (evaluación general del software incluyendo las limitaciones del mismo)
Firmas y aprobaciones de quienes deban supervisar el informe

PRUEBA DE GRAFOS

NOTACION DE GRAFO FLUJO



COMPLEJIDAD CICLOMATICA

- Camino 1: 1- 2 - 3 - 4 - 15
- Camino 2: 1- 2 - 3 - 5 - 6 - 14 - 15
- Camino 3: 1- 2 - 3 - 5 - 7 - 8 - 13 - 14 - 15
- Camino 4: 1- 2 - 3 - 5 - 7 - 9 - 10 - 12 - 13 - 14 - 15
- Camino 5: 1- 2 - 3 - 5 - 7 - 9 - 11 - 12 - 13 - 14 - 15

LA COMPLEJIDAD CICLOMATICA

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

$$V(G) = 18 - 15 + 2$$

$$V(G) = 5$$

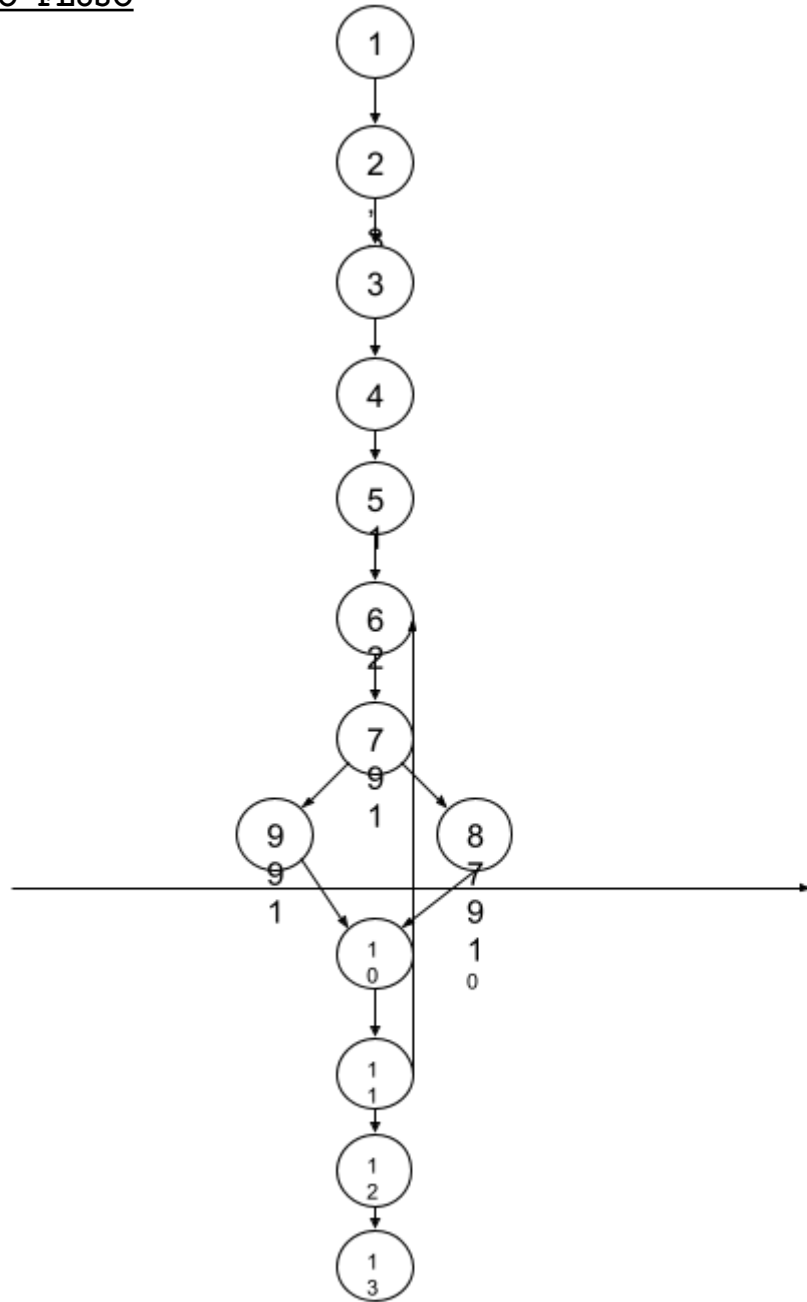
A = ARISTAS

N = NODOS

$$V(G) = 4 \text{ NODOS PREDICADO} + 1$$

$$V(G) = 5$$

NOTACION DE GRAFO FLUJO



COMPLEJIDAD CICLOMATICA

Camino 1: 1- 2 - 3 - 4 - 5 - 6 - 7 - 8 - 10 - 11 - 12 - 13

Camino 2: 1- 2 - 3 - 4 - 5 - 6 - 7 - 9 - 10 - 11 - 12 - 13

LA COMPLEJIDAD CICLOMATICA

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

$$V(G) = 15 - 13 + 2$$

$$V(G) = 4$$

A = ARISTAS

N = NODOS

$$V(G) = 3 \text{ NODOS PREDICADO} + 1$$

$$V(G) = 4$$

PRUEBA DE UNIDAD

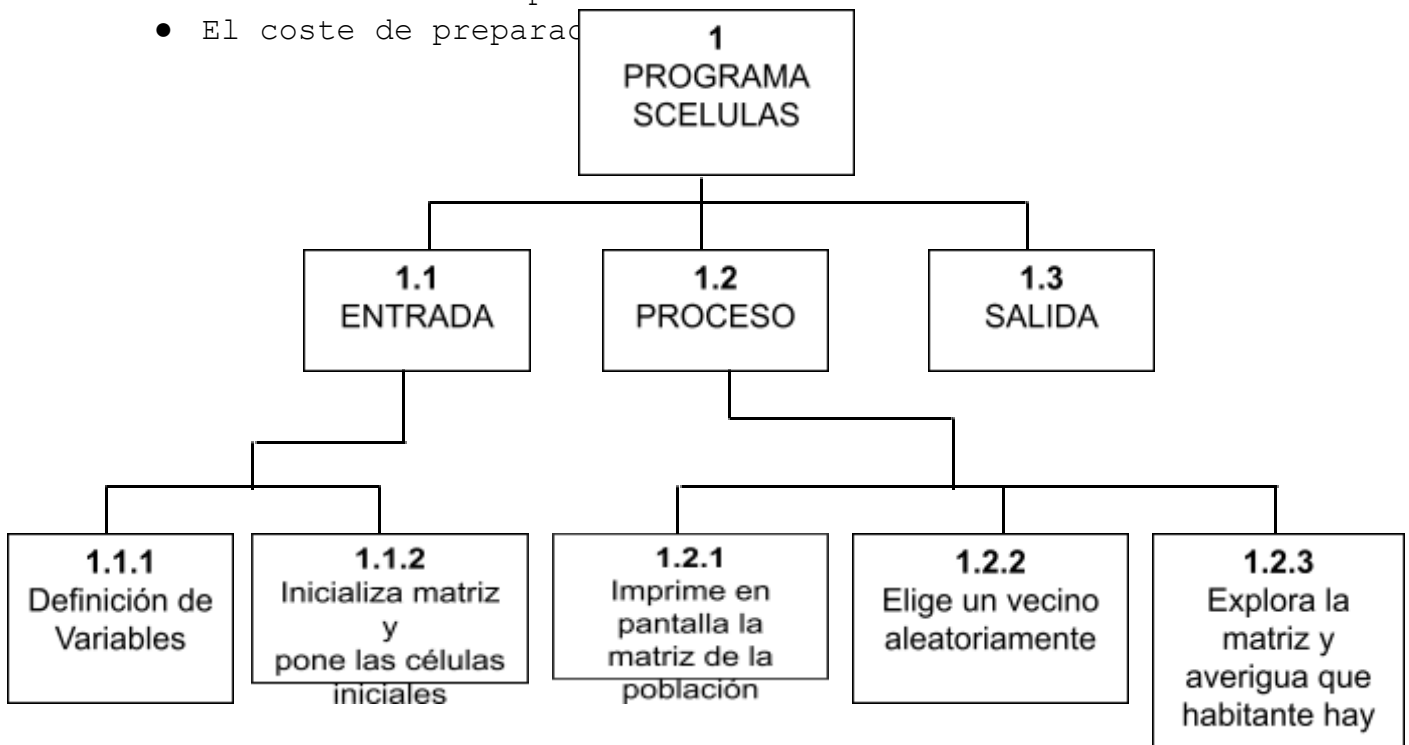
Hablamos de una unidad de prueba para referirnos a uno o más módulos que cumplen las siguientes condiciones [IEEE, 1986a]:

- Todos son del mismo programa
- Al menos uno de ellos no ha sido probado
- El conjunto de módulos es el objeto de un proceso de prueba

PRUEBAS DE INTEGRACION

Factores

- La forma de preparar casos
- Las herramientas necesarias
- El orden de codificar y probar los módulos
- El coste de la depuración
- El coste de preparar



PRUEBA DEL SISTEMA

- Cumplimiento de todos los requisitos funcionales, considerando el producto software final al completo en un entorno de sistema
- El funcionamiento y rendimiento en las interfaces hardware, software, de usuario y de operador
- Adecuación de la documentación de usuario
- Ejecución y rendimiento en condiciones límite y de sobrecarga

FUENTES DE DISEÑO DE CASOS DE PRUEBA

- Casos basados en los requisitos gracias a técnicas de caja negra aplicadas a las especificaciones
- Casos necesarios para probar el rendimiento del sistema y de su capacidad funcional (pruebas de volumen de datos, de límites de procesamiento, etc.).
- Este tipo de pruebas suelen llamarse pruebas de sobrecarga (stress testing)
- Casos basados en el diseño de alto nivel aplicando técnicas de caja blanca a los flujos de datos de alto nivel (por ejemplo, de los diagramas de flujo de datos)

Los elementos a ser probados son:

Pruebas de caja negra:

Áreas de prueba más importantes en el plan:

- Grafos
- Resistencia
- unidad
- integración
- interfaz grafica
- documentación y ayuda
- Tiempo real