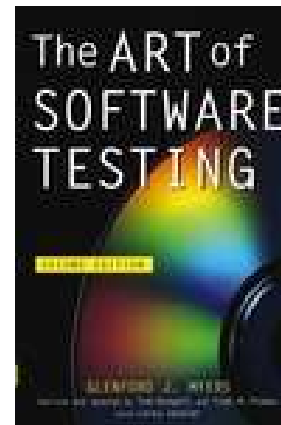


El arte de probar el software – Myers



Cátedra Ingeniería de Software

Ingeniería en Sistemas de Información - UTN – F. R. Rosario

Objetivos de esta presentación:

- La psicología de la prueba
- La economía de la prueba

Referencias

- El arte de probar el software – Myers - 1979
- The Art of Software Testing – Myers - 2004

La psicología de la prueba

Una de las causas principales de las pruebas del programa pobres es el hecho de que la mayoría de los desarrolladores comienzan con una falsa definición del término prueba.

Falsa definición del término prueba

"La prueba es el procedimiento para demostrar que los errores no están presentes."

"El propósito de las pruebas es demostrar que un programa realiza correctamente las funciones previstas."

"Las pruebas son el proceso de establecimiento de la confianza que un programa hace lo que se supone que debe hacer."

Correcta definición del término prueba

La prueba es el proceso de ejecutar un programa con la intención de encontrar errores

La economía de la prueba

Las pruebas exhaustivas de caja negra o caja blanca son imposible de realizar

La economía de la prueba

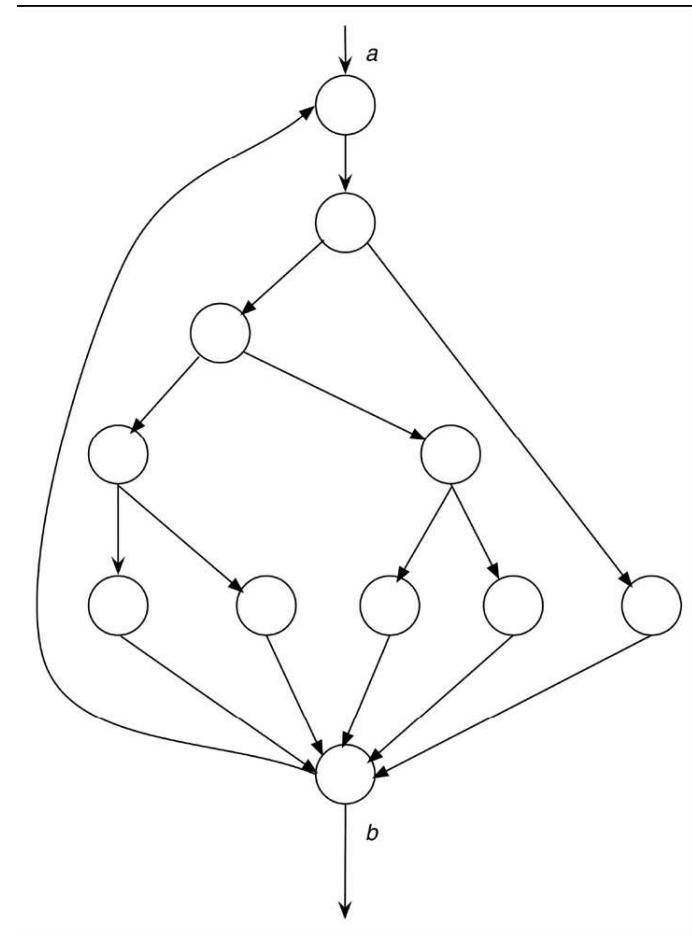
Pruebas tipo caja negra

- Para probar exhaustivamente el programa del triángulo habría que crear casos de prueba válidos y posibles. Debería usar un número de casos de prueba infinito.
- Dos implicaciones de esto son que:
 - no se puede probar un programa para garantizar que esté libre de errores
 - una consideración fundamental en la prueba de programas es la de la economía.

La economía de la prueba

Pruebas tipo caja blanca

- Grafico de control de un programa pequeño
- El número de secuencias lógicas distintas es aproximadamente 100 trillones



Principios de prueba

- 1 - Una parte necesaria de un caso de prueba es una definición de la salida prevista o resultado.
 - Por lo tanto, un caso de prueba deberá constar de dos componentes:
 - Una descripción de los datos de entrada al programa.
 - Una descripción precisa de la salida correcta del programa para este conjunto de datos de entrada.

Principios de prueba

- 2 - Un desarrollador debe evitar tratar de poner a prueba su propio programa.
 - Programar mirada constructiva y probar mirada destructiva
 - Falta de comprensión de la especificación
 - La prueba se hace con mayor efectividad si la realiza otra persona

Principios de prueba

3 - El personal de prueba no debería depender del área de desarrollo. (Una organización de programación no debería probar sus propios programas.)

Principios de prueba

4 - Inspeccione concienzudamente los resultados de cada prueba

Principios de prueba

5 - Los casos de prueba deben ser escritos tanto para las condiciones de entrada inválidas e inesperadas, como para condiciones válidas y esperadas.

Principios de prueba

6 - Examinar un programa para comprobar que no hace **lo que se supone** que debe hacer es sólo la mitad del problema. La otra mitad consiste en ver si el programa hace **lo que no se supone** que debe hacer.

- Los programas deben ser examinados por los efectos secundarios no deseados.
 - Por ejemplo, un programa para pago de sueldos que produce los cheques de pago correcto sigue siendo un programa erróneo si se produce también cheques adicionales para los empleados que no existen, o si escribe encima del primer registro del archivo de personal.

Principios de prueba

7 - Evite los casos de prueba desechables a menos que el programa sea verdaderamente un programa desechable.

- Inventar casos de prueba sobre la marcha y no documentarlos, genera una perdida de mucho trabajo.
- Cuando haya una mejora y deba probarse nuevamente el programa, los casos de prueba deben ser reinventados, y generalmente se hacen pruebas menos rigurosas, y si la modificación causa un error que antes no tenía, a menudo pasa desapercibido.
- Guardar los casos de prueba y ejecutarlos de nuevo después de cambiar otros componentes del programa se conoce como **pruebas de regresión**.

Principios de prueba

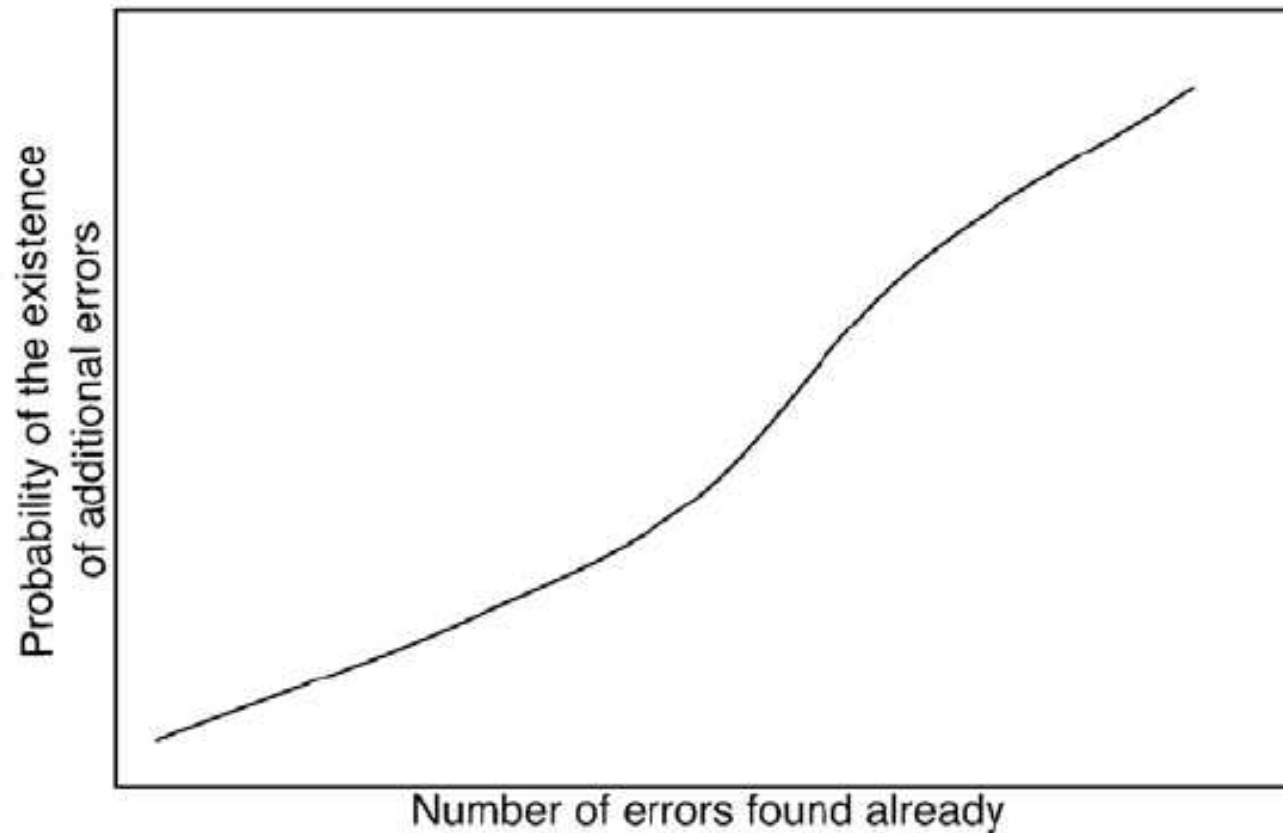
- 8 - No se debe planear el esfuerzo de pruebas, con la suposición tácita de que no se encontrarán errores.
- Los jefes de proyecto (project managers) cometen el error de suponer que la prueba es el proceso de mostrar que el programa funciona correctamente.
 - Prueba es el proceso de ejecución de un programa con la intención de encontrar errores.

Principios de prueba

9 - La probabilidad de encontrar errores adicionales en una sección de un programa es proporcional al número de errores ya encontrados en esa misma sección.

- Otra forma es decir que los errores tienden a aparecer en grupos y que algunas secciones parecen ser mucho más propensas a errores que otras secciones

Principios de prueba



- La sorprendente relación entre los errores remanentes y los errores encontrados

Principios de prueba

10 - Las pruebas constituyen una tarea altamente creativa y son un desafío intelectual

- La creatividad requerida para probar programas extensos supera la necesaria para concebirlos
- Existen métodos que permiten elaborar un conjunto razonable de casos de prueba para un programa, pero requieren un grado considerable de creatividad.

Resumen de los principios de prueba

Prueba es el proceso de ejecutar un programa con la intención de encontrar errores.

Un buen caso de prueba es el que tiene una probabilidad elevada de encontrar un error aún no descubierto.

Un caso de prueba exitoso es el que descubre un error, aún no descubierto.

Preguntas?

