

Metodología de la Programación (12016).

Algunos datos.

Dpto. de Informática e Ingeniería de Sistemas del Centro Politécnico Superior
Área de Lenguajes y Sistemas Informáticos
Universidad de Zaragoza, España

`ftricas@unizar.es`

`http://www.cps.unizar.es/~ftricas/`

Cifras

- ▶ El 30 % de los proyectos en entornos empresariales se cancelan sin haber sido finalizados.
- ▶ De los que se terminan, el 30 % cuesta al final entre un 150 % y un 200 % del presupuesto original.
- ▶ Menos del 10 % de proyectos en empresas grandes terminan a tiempo, y cumpliendo el presupuesto.

Más cifras

- ▶ Las tasas de defectos en productos comerciales se estiman entre 10 y 17 por cada 1000 líneas de código.
- ▶ Los estudios de la Universidad de Wisconsin muestran que más del 40 % de los programas populares en Windows pueden colgarse o interrumpirse con entradas generadas aleatoriamente.
- ▶ En sistemas de tipo Unix comerciales la proporción no es mejor.

Historias para no dormir

Algunos programas defectuosos son peligrosos:

'Un dispositivo de radioterapia controlado por computador (Therac-25) se vió implicado en al menos seis incidentes entre 1985 y 1987. Los pacientes fueron expuestos a grandes dosis de radiación que les causaron la muerte o daños importantes. El incidente se achaca a fallos de programación, aumentados por un diseño poco adecuado del sistema y de los procedimientos de gestión.'

Más historias para no dormir

Otros, cuando menos, caros:

'Investigadores de The Standish Group International estiman que el coste para las empresas norteamericanas de los fallos de los programas cuesta alrededor de 100 mil millones de dólares en pérdida de productividad y reparaciones'

Algunos fallos recientes

- ▶ 1990. Red telefónica de AT&T: 9 horas de interrupción por un error de programación.
- ▶ 1994. Pentium: error en la división en coma flotante costó más de 200 millones a Intel.
- ▶ 1996. Ariane 5: explosión aproximadamente 37 segundos después del despegue. El fallo fue debido a un error de conversión de datos en el paso de parámetros entre diferentes subsistemas.

Más fallos recientes

- ▶ 1999. Un defecto del sistema operativo provocó la corrupción en una base de datos importante, provocando que el sitio eBay.com interrumpiera su servicio durante 22 horas.
- ▶ Más en:
Collection of Software Bugs.

<http://www5.in.tum.de/~huckle/bugse.html>

Uno reciente

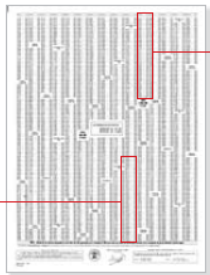
¡Lotería, lotería!

La lista de la lotería errónea

Del 54.501...

54501 ..c..1000
54502 ..c..1000
54503 ..c..1000
54504 ..c..1000
54505 ..c..1000
54506 ..c..1000
54507 ..c..1000
54508 ..c..1000
54509 ..c..1000
54510 ..c..1000
54511 ..c..1000
54512 ..c..1000
54513 ..c..1000

54543 ..c..1000
54544 ..c..1000
54545 ..c..1000
54546 ..c..1000
54547 ..c..1000
54548 ..c..1000
54549 ..c..1000
54550 ..c..1000
54551 ..c..1000
54552 ..c..1000
54553 ..c..1000



La primera lista oficial que distribuyó el Organismo Nacional de Loterías y Apuestas del Estado daba como premiados a 100 números por error. La centena premiada del Gordo abarcaba desde el 54.501 al 54.599 cuando en realidad debía hacerlo desde el 54.601 al 54.699.

El error, debido a un fallo informático, se conoció a las 15.00 y se publicó una nueva lista a las 22.00, hora en la que muchos diarios ya la habían impreso en sus ediciones especiales.

Números	Euros
54554 ..c..1000	
54555 ..c..1000	
54556 ..c..1000	
54557 ..c..1000	
54558 ..c..1000	
54559 ..c..1000	
54560 ..c..1000	
54561 ..c..1000	
54562 ..c..1000	
54563 ..c..1000	
54564 ..c..1000	

54591 ..c..1000	
54592 ..c..1000	
54593 ..c..1000	
54594 ..c..1000	
54595 ..c..1000	
54596 ..c..1000	
54597 ..c..1000	
54598 ..c..1000	
54599 ..c..1000	
54599 ..a..20000	

...al 54.599

Otro mas reciente

Miércoles 30/08/2006

ELMUNDO.ES

MADRID.- Todos los dominios '.es' desaparecieron a consecuencia de un "error de software" en los servidores DNS de la entidad pública ESNIC, responsable de la gestión del Registro de nombres de dominio de Internet bajo el código de país '.es'. Ningún sitio web con esta extensión fue accesible durante alrededor de dos horas y media en Internet

<http://www.elmundo.es/navegante/2006/08/29/tecnologia/1156867527.html>

El error se produjo entre las 15.15 y las 17.12 horas

Puede haber castigo

Cada vez se habla más de la responsabilidad de las empresas que desarrollan programas:

- ▶ 1999. Ambrosia Software (Rochester, N.Y.) anunció que si alguno de sus productos requerían la reparación de errores, el responsable de marketing comería insectos en alguna feria.

http://www.ambrosiasw.com/PRs/eatbugs_PR.html

Parece que finalmente tuvieron que comerlos . . .

http://www.ambrosiasw.com/news/old_newsletter.php?id=34019&page=3

- ▶ 31 de diciembre de 1999. Las autoridades chinas obligaron a los ejecutivos de la compañía aérea nacional a volar durante esa noche en los vuelos programados.

Historias con final feliz

El programa que se utiliza para controlar la lanzadera espacial de la NASA se ha desarrollado utilizando métodos formales y especificación. Un defecto puede ocasionar la muerte de seis astronautas y la pérdida de un equipo muy costoso.

Historias con final feliz (sigue)

- ▶ En marzo de 1997 el programa tenía 420000 líneas de código.
- ▶ La especificación ocupaba alrededor de 40000 páginas.
- ▶ Un cambio en el programa de navegación (menos del 2% del código)



2500 páginas de especificaciones antes de programar una sola línea de código.

Historias con final feliz (fin)

- ▶ Los programadores encontraron el 85 % de los errores antes del periodo de pruebas, y el 99.9 % antes de entregar el proyecto.
- ▶ Solamente ha aparecido un defecto por cada una de las tres últimas versiones.
- ▶ En las últimas 11 versiones el número total de fallos ha sido de 17, una media de 0.004 fallos por 1000 líneas de código.
- ▶ No es una excepción. Hay otras empresas que también son capaces de hacerlo.

Ejercicio

Suponer que A es un vector de enteros (que puede contener valores duplicados) y n un entero no negativo; escribir un programa que determine $nDist$, el número de valores diferentes almacenados en A , entre los subíndices de 1 a n , ambos incluidos.

El código no puede cambiar n ni A . El código no debería contener ineficiencias evidentes; sin embargo, podemos suponer que n es suficientemente pequeño como para que no valga la pena ordenar el vector (o una copia del vector).

¿Solución?

- ▶ ¿Está bien el programa sin probarlo en un computador?
- ▶ ¿Funciona bien cuando $n=0$?
- ▶ ¿Y cuando $n=1$?
- ▶ ¿Y si todos los elementos del vector son iguales?
- ▶ ¿Subiría a un avión controlado por un programa que use ese código?

Referencias

Casi todos los datos obtenidos en la introducción de:
R.D. Tennent; Specifying Software. A Hands-On Introduction.
2002. ISBN: 0-521-004012