Sesión 7: Bugs y debugging

Camilo Espejo[1]

`camilo.espejo@utadeo.edu.co`

[1]Maestría en Modelado y Simulación
Departamento de Ciencias Básicas, Univerisdad Jorge Tadeo Lozano

13 de septiembre de 2014

UNIVERSIDAD CENTRAL

UTADEO
UNIVERSIDAD DE BOGOTÁ JORGE TADEO LOZANO

Curso optativo:
Programación para M&S

Sumario

Bugs en los programas

"*It has been just so in all of my inventions. The first step is an intuition, and comes with a burst, then difficulties arise — this thing gives out and [it is] then that "Bugs"— as such little faults and difficulties are called — show themselves and months of intense watching, study and labor are requisite before commercial success or failure is certainly reached.*"

Thomas Edison, 1878.

Bugs en los programas

"*It has been just so in all of my inventions. The first step is an intuition, and comes with a burst, then difficulties arise — this thing gives out and [it is] then that "Bugs"— as such little faults and difficulties are called — show themselves and months of intense watching, study and labor are requisite before commercial success or failure is certainly reached.*"

Thomas Edison, 1878.

In 1946, when Hopper was released from active duty, she joined the Harvard Faculty at the Computation Laboratory where she continued her work on the Mark II and Mark III. Operators traced an error in the Mark II to a moth trapped in a relay, coining the term bug. This bug was carefully removed and taped to the log book. Stemming from the first bug, today we call errors or glitches in a program a bug.

Tipos de *bugs*

Comportamientos inesperados, más info: ▸ BUG

- ▸ *Bugs* aritméticos: división por cero, overflow, underflow, pérdida de precisión por redondeo o algorítmos inestables.

Tipos de *bugs*

- ▶ *Bugs* aritméticos: división por cero, overflow, underflow, pérdida de precisión por redondeo o algorítmos inestables.
- ▶ *Bugs* lógicos: recursividad infinita, loops infinitos. Errores *Off-by-one*.

Tipos de *bugs*

- ▶ *Bugs* aritméticos: división por cero, overflow, underflow, pérdida de precisión por redondeo o algorítmos inestables.
- ▶ *Bugs* lógicos: recursividad infinita, loops infinitos. Errores *Off-by-one*.
- ▶ *Bugs* de sintaxis: Uso de operadores equivocados (= Vs. ==).

Tipos de *bugs*

Comportamientos inesperados, más info: ▸ BUG

- ▶ *Bugs* aritméticos: división por cero, overflow, underflow, pérdida de precisión por redondeo o algorítmos inestables.
- ▶ *Bugs* lógicos: recursividad infinita, loops infinitos. Errores *Off-by-one*.
- ▶ *Bugs* de sintaxis: Uso de operadores equivocados (= Vs. ==).
- ▶ *Bugs* de recursos: (▸ deref),
  - ▶ Null pointer dereference.
  - ▶ Using an uninitialized variable.
  - ▶ Using an otherwise valid instruction on the wrong data type (see packed decimal/binary coded decimal).
  - ▶ Access violations.
  - ▶ Resource leaks, where a finite system resource (such as memory or file handles) become exhausted by repeated allocation without release.
  - ▶ Buffer overflow, in which a program tries to store data past the end of allocated storage. This may or may not lead to an access violation or storage violation. These bugs can form a security vulnerability.
  - ▶ Excessive recursion which — though logically valid — causes stack overflow.
  - ▶ Use-after-free error, where a pointer is used after the system has freed the memory it references.
  - ▶ Double free error.

Debugging

- Encontrar y reparar los bugs.
- Es preciso automatizar el proceso $\rightarrow$ GDB