

## Mars Climate Orbiter

- Purpose: to relay signals from the Mars Polar Lander once it reached the surface of the planet
- Disaster: smashed into the planet instead of reaching a safe orbit
- Why: Software bug - failure to convert English measures to metric values
- \$165M



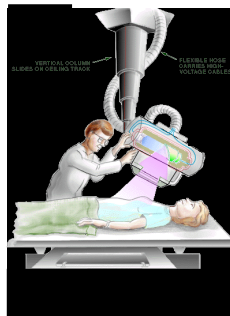
## Shooting Down of Airbus 320

- 1988
- US Vincennes shot down Airbus 320
- Mistook airbus 320 for a F-14
- 290 people dead
- Why: Software bug - cryptic and misleading output displayed by the tracking software



## THERAC-25 Radiation Therapy

- THERAC-25, a computer-controlled radiation-therapy machine
- 1986: two cancer patients at the East Texas Cancer Center in Tyler received fatal radiation overdoses
- Why: Software bug - mishandled race condition (i.e., miscoordination between concurrent tasks)



## London Ambulance Service

- London Ambulance Service Computer Aided Dispatch (LASCAD)
- Purpose: automate many of the human-intensive processes of manual dispatch systems associated with ambulance services in the UK
  - functions: Call taking
- Failure of the London Ambulance Service on 26 and 27 November 1992

## "Nice of You to Turn Up"

- Load increased
- Emergencies accumulated
- System made incorrect allocations
  - more than one ambulance being sent to the same incident
  - the closest vehicle was not chosen for the emergency
- At 23:00 on October 28 the LAS eventually instigated a backup procedure, after the death of at least 20 patients

## More...

- "Software and its Impact: A Quantitative Assessment," by B.W. Boehm, Datamation, 19(5), 48-59 (1973)
  - Errors in medical software have caused deaths

## More...

- "The development of software for ballistic-missile defense," by H. Lin, Scientific American, vol. 253, no. 6 (Dec. 1985), p. 48
  - British destroyer H.M.S. Sheffield; sunk in the Falkland Islands war; ship's radar warning system software allowed missile to reach its target
  - An Air New Zealand airliner crashed into an Antarctic mountain
  - North American Aerospace Defense Command reported that the U.S. was under missile attack; traced to faulty computer software - generated incorrect signals
  - Manned space capsule Gemini V missed its landing point by 100 miles; software ignored the motion of the earth around the sun

## More...

- "Software Engineering: Report on a Conference Sponsored by the NATO Science Committee, Brussels, NATO Scientific Affairs Division," 1968, p. 121
  - An error in an aircraft design program contributed to several serious air crashes
- "Ghost in the Machine," Time Magazine, Jan. 29, 1990. p. 58
  - Dallas/Fort Worth air-traffic system began spitting out gibberish in the Fall of 1989 and controllers had to track planes on paper

## More...

- Software Reliability: Principles & Practice, p. 25, by G. J. Myers
  - Apollo 8 spacecraft erased part of the computer's memory
  - Eighteen errors were detected during the 10-day flight of Apollo 14
  - An error in a single FORTRAN statement resulted in the loss of the first American probe to Venus

## More...

- An Airbus A320 crashes at an air show
- A China Airlines Airbus Industries A300 crashes on April 26, 1994 killing 264
- Ariane 5 satellite launcher malfunction was caused by a faulty software exception routine resulting from a bad 64-bit floating point to 16-bit integer conversion

## More...

- ACM SIGSOFT Software Engineering Notes, vol. 6, no. 2
  - F-18 fighter plane crashed due to a missing exception condition
- ACM SIGSOFT Software Engineering Notes, vol. 9, no. 5
  - F-14 fighter plane was lost to uncontrollable spin, traced to tactical software

## More...

- Internet Risks Forum NewsGroup (RISKS), vol. 19, issue 56
  - CyberSitter censors "menu \*/ #define" because of the string "nu...de"
- London's Docklands Light Railway - train stopped in the middle of nowhere due to future station location programmed in software
- ACM SIGSOFT Software Engineering Notes, vol. 12, no. 3
  - Chicago cat owners were billed \$5 for unlicensed dachshunds. A database search on "DHC" (for dachshunds) found "domestic house cats" with shots but no license



More...

- and many many more ....



CNN.com

PRINT THIS  
Powered by clickability

## Russia: Software bug made Soyuz stray

STAR CITY, Russia (AP) --A computer software error likely sent a Russian spacecraft into a rare ballistic descent that subjected the three men on board to chest-crushing gravity loads that made it hard to breathe, space experts said Tuesday.

"For me, for a moment, it felt like I was Atlas and I had the weight of the whole world on my shoulders," astronaut Donald Pettit, still a little woozy, told reporters at a crowded news conference.

CNN.com

PRINT THIS  
Powered by clickability

## Spread of buggy software raises new questions

NEW YORK (AP) --When his dishwasher acts up and won't stop beeping, Jeff Seigle turns it off and then on, just as he does when his computer crashes. Same with the exercise machines at his gym and his CD player.

"Now I think of resetting appliances, not just computers," says Seigle, a software developer in Vienna, Virginia.

Malfunctions caused by bizarre and frustrating glitches are becoming harder and harder to escape now that software controls everything from stoves to cell phones, trains, cars and power plants.

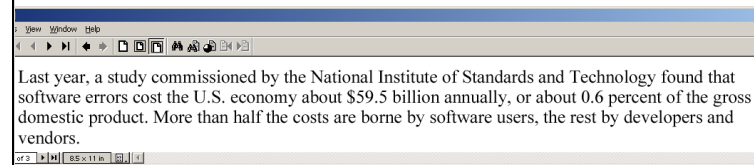
--A poorly programmed ground-based altitude warning system was partly responsible for the 1997 Korean Air crash in Guam that killed 228 people.

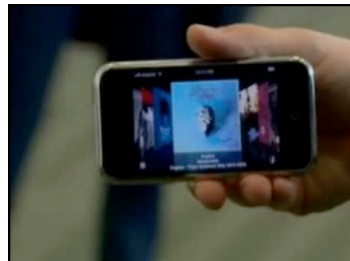
--Faulty software in anti-lock brakes forced the recall of 39,000 trucks and tractors and 6,000 school buses in 2000.

--The \$165 million Mars Polar Lander probe was destroyed in its final descent to the planet in 1999, probably because its software shut the engines off 100 feet above the surface.

## Economic Impact

- NIST study
- On CNN.com - April 27, 2003





Open up albums.  
Browse through the albums in  
horizontal mode.



Select an album



Play song from the  
selected album

Browse again  
through the albums



Get back to vertical mode.



Now see the browser get back  
to the song that was played in  
vertical mode not the current screen  
that the user were browsing

## Expectation

- Can't we expect software to execute correctly?
- Carefully made programs
  - 5 faults/1000 LOC
  - 1M LOC will have 5000 faults
- Windows XP has 45M LOC
  - How many faults?
  - $45 \times 5000 = 225,000$
- Why not remove the faults?

```

*** STOP: 0x0000000A (0x00000000,0x00000002,0x00000000,8038c240)
IRQL_NOT_LESS_OR_EQUAL*** Address 8038c240 has base at 8038c000 - Ntfs.SYS

CPUID: Genuine Intel 6.3.3 irql:1f SYSVER 0xf00005e5

```

Dll Base	DateStamp	Name	Dll Base	DateStamp	Name
80100000	336546bf	- ntoskrnl.exe	80010000	33247f88	- hal.dll
80000100	334d3a53	- atapi.sys	80007000	33248048	- SCSIPORT.SYS
802aa000	33013a6b	- epst.mpd	802b5000	336016a2	- Disk.sys
802b9000	336015af	- CLASS2.SYS	8038c000	3356d637	- Ntfs.sys
802b4000	334844be	- Siuvvid.sys	803e4000	33d84553	- Ntice.sys
f9318000	31ec6c8d	- Floppy.SYS	f95c9000	31ac6c99	- Null.SYS
f9468000	31ed368b	- KSecDD.SYS	f95ca000	335e0cf	- Beep.SYS
f9358000	335bc82a	- i8042prt.sys	f9474000	3324806f	- mouclass.sys
f947c000	31ec6c94	- kbdclass.sys	f95cb000	3373c39d	- ctrl2cap.SYS
f9370000	33248011	- VIDEOPORT.SYS	fe9d7000	3370e7b9	- ati.sys
f9490000	31ec6c6d	- vga.sys	f93b0000	332480dd	- Mfs.SYS
f90f0000	332480d0	- Npfs.SYS	fe957000	3356da41	- NDIS.SYS
a0000000	335157ac	- win32k.sys	fe914000	334ea144	- ati.dll
fe0c9000	335bd30e	- Fastfat.SYS	fe110000	31ec7c9b	- Parport.SYS
fe108000	31ec6c9b	- Parallel.SYS	f95b4000	31ec6c9d	- ParVdm.SYS
f9050000	332480ab	- Serial.SYS			

```

Address  dword  dump  Build [1314]          - Name
801afc24  80149905  80149905  ff8e6b8c  80129c2c  ff8e6b94  8025c000  - Ntfs.SYS
801afc2c  80129c2c  80129c2c  ff8e6b94  00000000  ff8e6b94  80100000  - ntoskrnl.exe
801afc34  801240f2  801240f2  ff8e6d14  ff8e6f60  ff8e6c58  80100000  - ntoskrnl.exe
801afc54  80124f16  80124f16  ff8e6f60  ff8e6c3c  8015ac7e  80100000  - ntoskrnl.exe
801afc64  8015ac7e  8015ac7e  ff8e6d14  ff8e6f60  ff8e6c58  80100000  - ntoskrnl.exe
801afc70  80129bda  80129bda  00000000  80088000  80106f60  80100000  - ntoskrnl.exe

```

Restart and set the recovery options in the system control panel or the /CRASHDEBUG system start option. If this message reappears, contact your system administrator or technical support group.

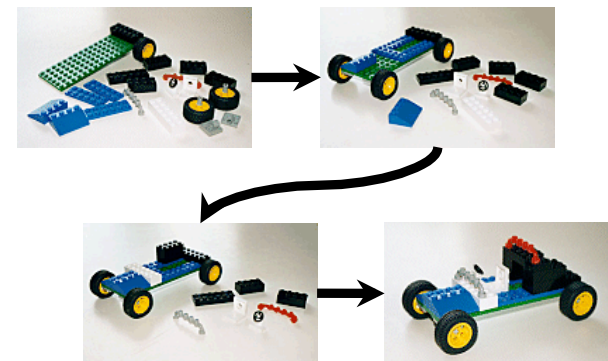
## Joke?

- "If the automobile industry had developed like the software industry, we would all be driving \$25 cars that get 1,000 miles to the gallon."
- "Yeah, and if cars were like software, they would crash twice a day for no reason, and when you called for service, they'd tell you to reinstall the engine."

## How Cars Are Engineered (A Simple View)

- User requirements
  - Engine power, all-wheel, seating, comfort, MP3 player!!
- Detailed design
  - Blueprints, design documents
- Verify design
  - Simulation, prototyping
- Develop parts (components)
  - Test each component
  - Components may be reused
  - Mass produced
- Assemble the car
  - Test the car (Front/side crash tests, Stability tests)
  - Usability testing (Feedback from drivers/passengers)

## How Cars Are REALLY Engineered (A Detailed View)



## But Seriously



- Features of many LEGO parts
  - Modularity
  - Reusability
    - Each part can be used in different places and ways
  - Flexibility of design
  - Compatibility
    - With other LEGO sets
- Building-blocks

## Goals of the Course

- Discuss software testing techniques
- Two parts of the course
  - Review testing fundamentals
  - State-of-the-art & emerging techniques
- What do I expect from students?

## MS and Ph.D. Qualifying

- Is the course is valid for PhD qualifying coursework?
  - Yes (Software Engineering/Programming Languages)
- Is the course is valid for MS qualifying coursework?
  - Yes (Software Engineering/Programming Languages)
- Is the course is valid for MS comps?
  - Yes (Both Midterm and Final exams count towards the MS comps.)

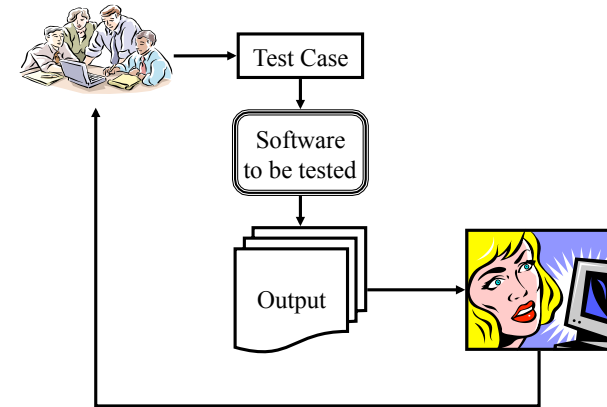
## Assessment

- 25% Mid-term Exam
- 25% Final Exam
- 50% Project
  - Project report
  - Project Presentation

## Exam Contents

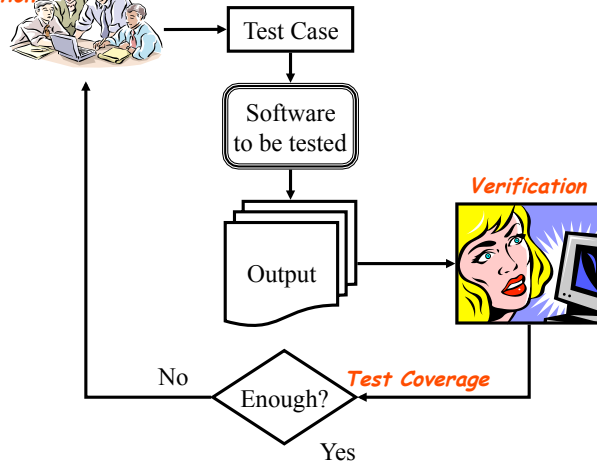
- Midterm
  - Everything discussed in class
- Final exam
  - Everything discussed/presented after midterm

## Testing: Our Experiences

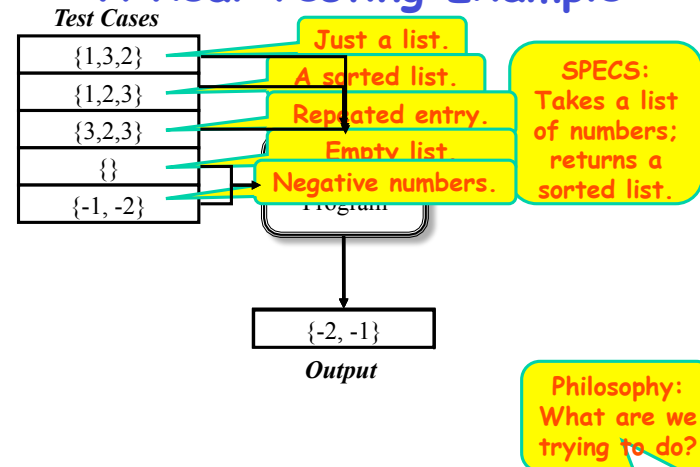


## When to Stop?

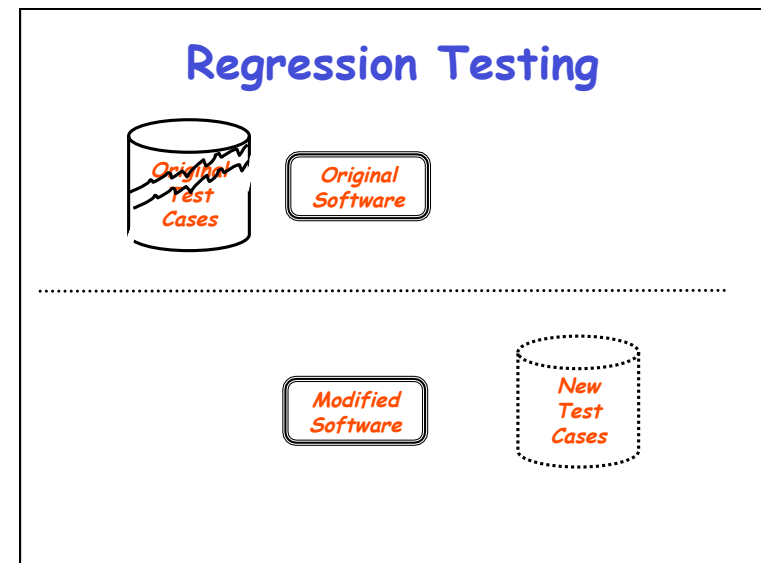
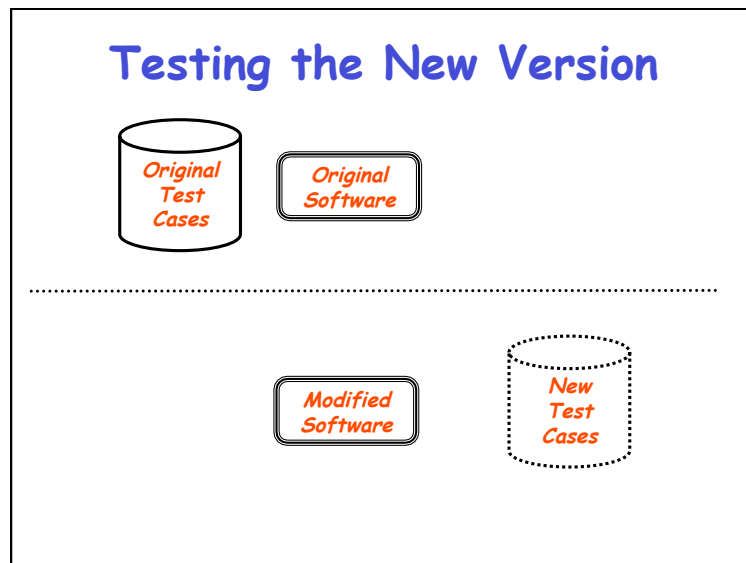
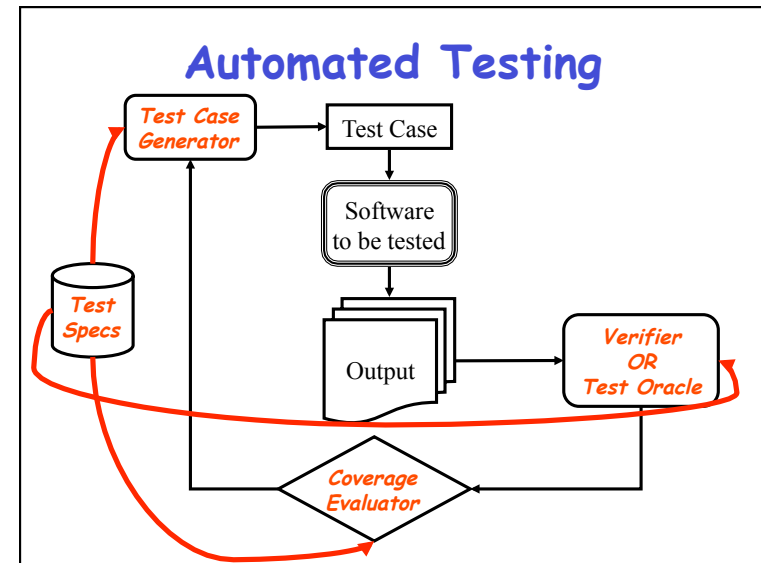
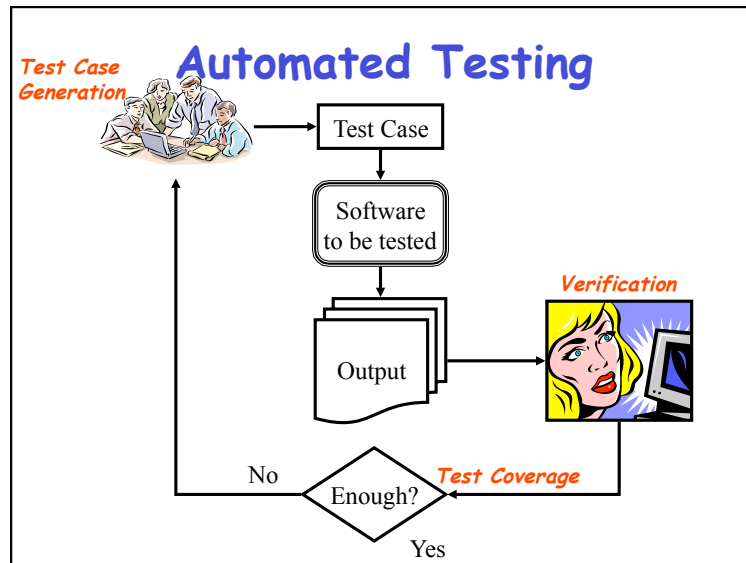
Test Case Generation



## A Real Testing Example







## Discussion

- Different Software Types
  - Object-oriented
  - Component-based
  - Concurrent
  - Distributed
  - Graphical-user Interfaces
  - Web
- Different goals of testing
  - Usability
  - Security
  - Correctness
  - Performance ...