CS1632, Lecture 6

Wonsun ahn

Software tends not to break much on the “happy path”

- Happy Path: a case where user inputs valid, usual values; expected usage
- It breaks on the unexpected cases.
  - Corner cases.
  - Systems going down.
  - Malicious users.
  - When you're off in the wilderness.

Logic Errors:
The logic of the program is incorrect

```
if (student.isTaking(cs1632)) {
   student.setHappy(false);
} else {
   student.setHappy(true);
}
```

OFF-by-one error:
a subset of logic errors where values are specified incorrectly by one unit

```
if (student.getNumCredits() > 120) {
   student.setCanGraduate(true);
} else {
   student.setCanGraduate(false);
}
```

Rounding/floating point errors: rounding or floating point display give incorrect results.

```
double oneVal = 1.0 / 857.0;
double total = oneVal * 857.0;

System.out.println("Should be 1.0, actually = " + total);

boolean areEqual = (total == 1.0);
System.out.println("Are equal? " + areEqual);
```

Integration errors:
Errors at boundaries between systems/subsystems.

```
int startDistanceInKilometers = 14;
spacecraft.setDistance(startDistanceInKilometers);

...

public class Spacecraft
   public void setDistance(int distanceInMiles) {
      ...
   }
}
```

Errors of assumption:
developer or system makes an assumption which turns out to be incorrect, or at odds with other assumptions.

```
OutputFile.write(TAB_DELIMITED);

...

InputFile.read(COMMA_DELIMITED);
```

MISSING DATA ERRORS:

An error occurs because needed data is missing and the system cannot operate properly without it.

```
public static void main(String[] args) {
    System.out.println(args[3]);
}
```

BAD DATA ERRORS:

System cannot handle improperly formatted or invalid data.

```
Enter two numbers to divide: 7 0
Exception in thread "main“
    java.lang.ArithmeticException: / by zero
```

DISPLAY ERRORS:

The data is correct but not displayed properly.

```
double pi = Math.PI;
System.out.printf("Pi is equal to...%.1f!",
    pi);
```

Null pointer error:
The program dereferences a null pointer.

```
String oneILove = null;
oneILove = oneILove.toUpperCase();

System.out.printf("This one goes out to
the one I love," + oneILove);
```

I/O Errors:

The system encounters an unexpected state of disk, network, or other I/O and cannot handle it.

```
try {
    // read in file
} catch (FileNotFoundException e) {
    // AAARGH WHAT DO I DO
    System.exit(1);
}
```

Configuration error:
The system could work correctly, but it was not configured to work correctly.

```
'javac' is not recognized as an internal
or external command, operable program or
batch file.
```

The list goes on…

- Permission errors
- Version mismatch errors
- Distributed system errors
- Interface errors
- Domain-specific errors