Breaking Software

Software tends not to break much on the happy path.

It breaks when there's unexpected input.
Malicious users.
Systems going down.

When you're off in the wilderness.

Off-By-One Errors

[illegible]

Logic Errors
The logic of the program is incorrect.

```
if (age < 35) {
    System.out.println("You can president!");
} else {
    System.out.println("Wait a few years...");
}
```

Rounding / Floating Point Errors

[illegible]

Prezi

Breaking Software

Software tends not to break much on the happy path.

It breaks when there's unexpected input.
Malicious users.
Systems going down.

When you're off in the wilderness.

Off-By-One Errors

A subset of logic errors where values are specified incorrectly by one unit.

```
if (age > 35) {
    System.out.println("You can president!");
} else {
    System.out.println("Wait a few years...");
}
```

Logic Errors

The logic of the program is incorrect.

```
if (age < 35) {
   System.out.println("You can president!");
} else {
   System.out.println("Wait a few years...");
}
```

Rounding / Floating Point Errors

Rounding or floating point give incorrect results.

```
double oneVal = 1.0 / 557.0;
double total = oneVal * 557.0;
System.out.println("Should be 1.0, actually = " + total);
boolean areEqual = (total == 1.0);
System.out.println("Are equal? " + areEqual);
```

Integration Errors

[illegible]

Prezi

Software tends not to break much on the happy path.

It breaks when there's unexpected input.
Malicious users.
Systems going down.

When you're off in the wilderness.

Logic Errors

The logic of the program is incorrect.

```
if (age < 35) {
   System.out.println("You can president!");
} else {
   System.out.println("Wait a few years...");
}
```

Off-By-One Errors

A subset of logic errors where values are specified incorrectly by one unit.

```
if (age > 35) {
   System.out.println("You can president!");
} else {
   System.out.println("Wait a few years...");
}
```

Rounding / Floating Point Errors

Rounding or floating point 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) {
   ...
}
```

Prezi

Errors of Assumption

The 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 exactly... %.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);
```

Network / File / I/O Error

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

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

Configuration Errors

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...

Accessibility errors...
Domain-specific errors...
Version mismatch errors...
Distributed system errors...
Interface errors...
Logging errors...
Regulatory errors...
etc. etc.

Prezi

Breaking Software

Software tends not to break much on the happy path.

It breaks when there's unexpected input.
Malicious users.
Systems going down.

When you're off in the wilderness.

Off-By-One Errors

A subset of logic errors where values are specified incorrectly by one unit.

```
if (age > 35) {
  System.out.println("You can president!");
} else {
  System.out.println("Wait a few years...");
}
```

Logic Errors

The logic of the program is incorrect.

```
if (age < 35) {
  System.out.println("You can president!");
} else {
  System.out.println("Wait a few years...");
}
```

Rounding / Floating Point Errors

Rounding or floating point give incorrect results.

```
double oneVal = 1.0 / 957.0;
double total = oneVal * 957.0;
System.out.println("Should be 1.0, actually = " + total);
boolean areEqual = (total == 1.0);
System.out.println("Are equal? " + areEqual);
```

Integration Errors

Prezi