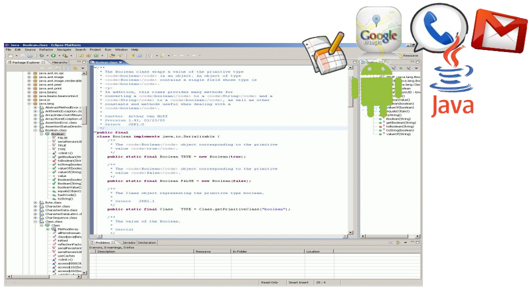
What is Software Engineering?

What is Software Engineering?

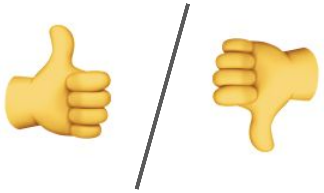
👍 / 👎

- Developing in an IDE
  and software ecosystem?

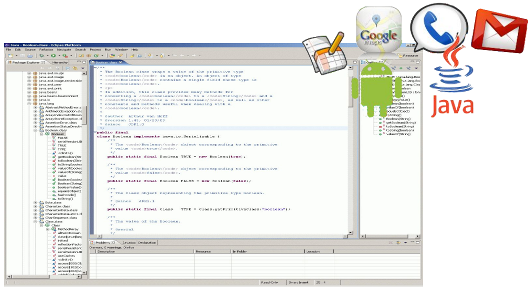


```
/**
 * The Boolean class wraps a value of the primitive type
 * <code>boolean</code> in an object. An object of type
 * <code>Boolean</code> contains a single field whose type is
 * <code>boolean</code>.
 * <p>
 * In addition, this class provides many methods for
 * converting a <code>boolean</code> to a <code>String</code> and a
 * <code>String</code> to a <code>boolean</code>, as well as other
 * constants and methods useful when dealing with a
 * <code>boolean</code>.
 *
 * @author  Arthur van Hoff
 * @version 1.43, 01/23/03
 * @since   JDK1.0
 */
public final
class Boolean implements java.io.Serializable {
    /**
     * The <code>Boolean</code> object corresponding to the primitive
     * value <code>true</code>.
     */
    public static final Boolean TRUE = new Boolean(true);

    /**
     * The <code>Boolean</code> object corresponding to the primitive
     * value <code>false</code>.
     */
    public static final Boolean FALSE = new Boolean(false);

    /**
     * The Class object representing the primitive type boolean.
     *
     * @since   JDK1.1
     */
    public static final Class   TYPE = Class.getPrimitiveClass("boolean");

    /**
     * The value of the Boolean.
     *
     * @serial
     */
```

What is Software Engineering?



- Developing in an IDE and software ecosystem?

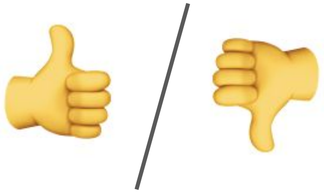


```
/**
 * The Boolean class wraps a value of the primitive type
 * <code>boolean</code> in an object. An object of type
 * <code>Boolean</code> contains a single field whose type is
 * <code>boolean</code>.
 * <p>
 * In addition, this class provides many methods for
 * converting a <code>boolean</code> to a <code>String</code> and a
 * <code>String</code> to a <code>boolean</code>, as well as other
 * constants and methods useful when dealing with a
 * <code>boolean</code>.
 *
 * @author  Arthur van Hoff
 * @version 1.43, 01/23/03
 * @since   JDK1.0
 */
public final
class Boolean implements java.io.Serializable {
    /**
     * The <code>Boolean</code> object corresponding to the primitive
     * value <code>true</code>.
     */
    public static final Boolean TRUE = new Boolean(true);

    /**
     * The <code>Boolean</code> object corresponding to the primitive
     * value <code>false</code>.
     */
    public static final Boolean FALSE = new Boolean(false);

    /**
     * The Class object representing the primitive type boolean.
     *
     * @since   JDK1.1
     */
    public static final Class   TYPE = Class.getPrimitiveClass("boolean");

    /**
     * The value of the Boolean.
     *
     * @serial
     */
```

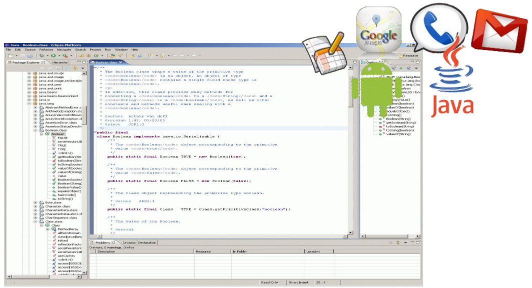
0 errors, 0 warnings, 0 infos

- Coding and debugging?

What is Software Engineering?



- Developing in an IDE and software ecosystem?

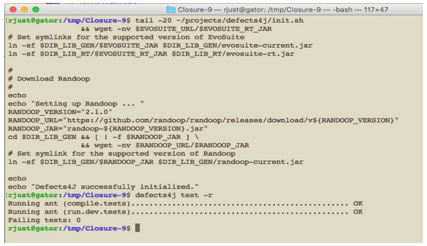


```
/**
 * The Boolean class wraps a value of the primitive type
 * <code>boolean</code> in an object. An object of type
 * <code>Boolean</code> contains a single field whose type is
 * <code>boolean</code>.
 * <p>
 * In addition, this class provides many methods for
 * converting a <code>boolean</code> to a <code>String</code> and a
 * <code>String</code> to a <code>boolean</code>, as well as other
 * constants and methods useful when dealing with a
 * <code>boolean</code>.
 *
 * @author  Arthur van Hoff
 * @version 1.43, 01/23/03
 * @since   JDK1.0
 */
public final
class Boolean implements java.io.Serializable {
    /**
     * The <code>Boolean</code> object corresponding to the primitive
     * value <code>true</code>.
     */
    public static final Boolean TRUE = new Boolean(true);

    /**
     * The <code>Boolean</code> object corresponding to the primitive
     * value <code>false</code>.
     */
    public static final Boolean FALSE = new Boolean(false);

    /**
     * The Class object representing the primitive type boolean.
     *
     * @since   JDK1.1
     */
    public static final Class   TYPE = Class.getPrimitiveClass("boolean");

    /**
     * The value of the Boolean.
     *
     * @serial
     */
```

0 errors, 0 warnings, 0 infos

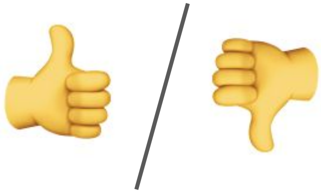
- Coding and debugging?

- Deploying and running a software system?

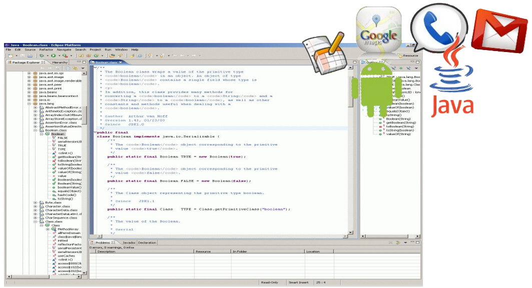


```
rjust@gator:/tmp/Closure-9$ tail -20 ~/projects/defects4j/init.sh
                && wget -nv $EVOSUITE_URL/$EVOSUITE_RT_JAR
# Set symlinks for the supported version of EvoSuite
ln -sf $DIR_LIB_GEN/$EVOSUITE_JAR $DIR_LIB_GEN/evosuite-current.jar
ln -sf $DIR_LIB_RT/$EVOSUITE_RT_JAR $DIR_LIB_RT/evosuite-rt.jar

#
# Download Randoop
#
echo
echo "Setting up Randoop ... "
RANDOOP_VERSION="2.1.0"
RANDOOP_URL="https://github.com/randoop/randoop/releases/download/v${RANDOOP_VERSION}"
RANDOOP_JAR="randoop-${RANDOOP_VERSION}.jar"
cd $DIR_LIB_GEN && [ ! -f $RANDOOP_JAR ] \
                && wget -nv $RANDOOP_URL/$RANDOOP_JAR
# Set symlink for the supported version of Randoop
ln -sf $DIR_LIB_GEN/$RANDOOP_JAR $DIR_LIB_GEN/randoop-current.jar

echo
echo "Defects4J successfully initialized."
rjust@gator:/tmp/Closure-9$ defects4j test -r
Running ant (compile.tests)................................................ OK
Running ant (run.dev.tests)................................................ OK
Failing tests: 0
rjust@gator:/tmp/Closure-9$ █
```

What is Software Engineering?

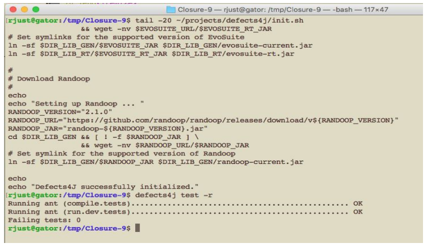


- Developing in an IDE and software ecosystem?



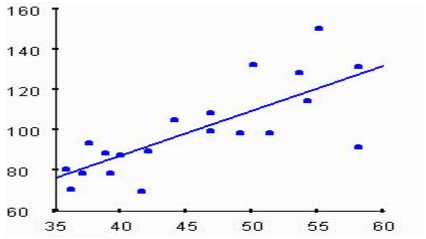
- Coding and debugging?

- Deploying and running a software system?



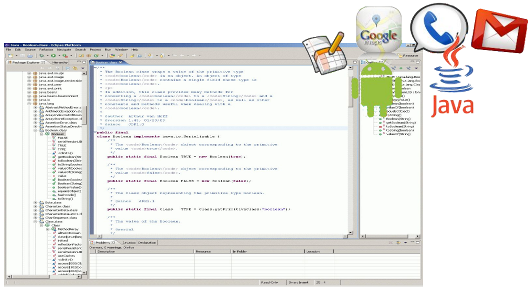
```
rjust@gator:/tmp/Closure-9$ tail -20 ~/projects/defects4j/init.sh
                && wget -nv $EVOSUITE_URL/$EVOSUITE_RT_JAR
# Set symlinks for the supported version of EvoSuite
ln -sf $DIR_LIB_GEN/$EVOSUITE_JAR $DIR_LIB_GEN/evosuite-current.jar
ln -sf $DIR_LIB_RT/$EVOSUITE_RT_JAR $DIR_LIB_RT/evosuite-rt.jar

#
# Download Randoop
#
echo
echo "Setting up Randoop ... "
RANDOOP_VERSION="2.1.0"
RANDOOP_URL="https://github.com/randoop/randoop/releases/download/v${RANDOOP_VERSION}"
RANDOOP_JAR="randoop-${RANDOOP_VERSION}.jar"
cd $DIR_LIB_GEN && [ ! -f $RANDOOP_JAR ] \
                && wget -nv $RANDOOP_URL/$RANDOOP_JAR
# Set symlink for the supported version of Randoop
ln -sf $DIR_LIB_GEN/$RANDOOP_JAR $DIR_LIB_GEN/randoop-current.jar

echo
echo "Defects4J successfully initialized."
rjust@gator:/tmp/Closure-9$ defects4j test -r
Running ant (compile.tests)............................................... OK
Running ant (run.dev.tests)............................................... OK
Failing tests: 0
rjust@gator:/tmp/Closure-9$ 
```

- Empirical evaluations?



What is Software Engineering?

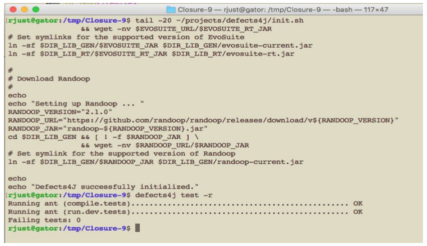
- Developing in an IDE and software ecosystem?


Eclipse Platform
Navigate Search Project Run Window Help
Hierarchy
Boolean.class
Problems Javadoc Declaration
0 errors, 0 warnings, 0 infos
Description Resource In Folder Location
Read-Only Smart Insert 25 : 4
Google maps
Java

```
/**
 * The Boolean class wraps a value of the primitive type
 * <code>boolean</code> in an object. An object of type
 * <code>Boolean</code> contains a single field whose type is
 * <code>boolean</code>.
 * <p>
 * In addition, this class provides many methods for
 * converting a <code>boolean</code> to a <code>String</code> and a
 * <code>String</code> to a <code>boolean</code>, as well as other
 * constants and methods useful when dealing with a
 * <code>boolean</code>.
 *
 * @author  Arthur van Hoff
 * @version 1.43, 01/23/03
 * @since   JDK1.0
 */
public final
class Boolean implements java.io.Serializable {
    /**
     * The <code>Boolean</code> object corresponding to the primitive
     * value <code>true</code>.
     */
    public static final Boolean TRUE = new Boolean(true);

    /**
     * The <code>Boolean</code> object corresponding to the primitive
     * value <code>false</code>.
     */
    public static final Boolean FALSE = new Boolean(false);

    /**
     * The Class object representing the primitive type boolean.
     *
     * @since   JDK1.1
     */
    public static final Class   TYPE = Class.getPrimitiveClass("boolean");

    /**
     * The value of the Boolean.
     *
     * @serial
     */
```

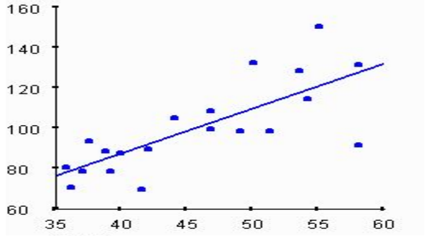
- Coding and debugging?

- Deploying and running a software system?



```
rjust@gator:/tmp/Closure-9$ tail -20 ~/projects/defects4j/init.sh
                && wget -nv $EVOSUITE_URL/$EVOSUITE_RT_JAR
# Set symlinks for the supported version of EvoSuite
ln -sf $DIR_LIB_GEN/$EVOSUITE_JAR $DIR_LIB_GEN/evosuite-current.jar
ln -sf $DIR_LIB_RT/$EVOSUITE_RT_JAR $DIR_LIB_RT/evosuite-rt.jar

#
# Download Randoop
#
echo
echo "Setting up Randoop ... "
RANDOOP_VERSION="2.1.0"
RANDOOP_URL="https://github.com/randoop/randoop/releases/download/v${RANDOOP_VERSION}"
RANDOOP_JAR="randoop-${RANDOOP_VERSION}.jar"
cd $DIR_LIB_GEN && [ ! -f $RANDOOP_JAR ] \
                && wget -nv $RANDOOP_URL/$RANDOOP_JAR
# Set symlink for the supported version of Randoop
ln -sf $DIR_LIB_GEN/$RANDOOP_JAR $DIR_LIB_GEN/randoop-current.jar

echo
echo "Defects4J successfully initialized."
rjust@gator:/tmp/Closure-9$ defects4j test -r
Running ant (compile.tests)................................................ OK
Running ant (run.dev.tests)................................................ OK
Failing tests: 0
rjust@gator:/tmp/Closure-9$ 
```

- Empirical evaluations?



All of the above -- much more than just writing code!

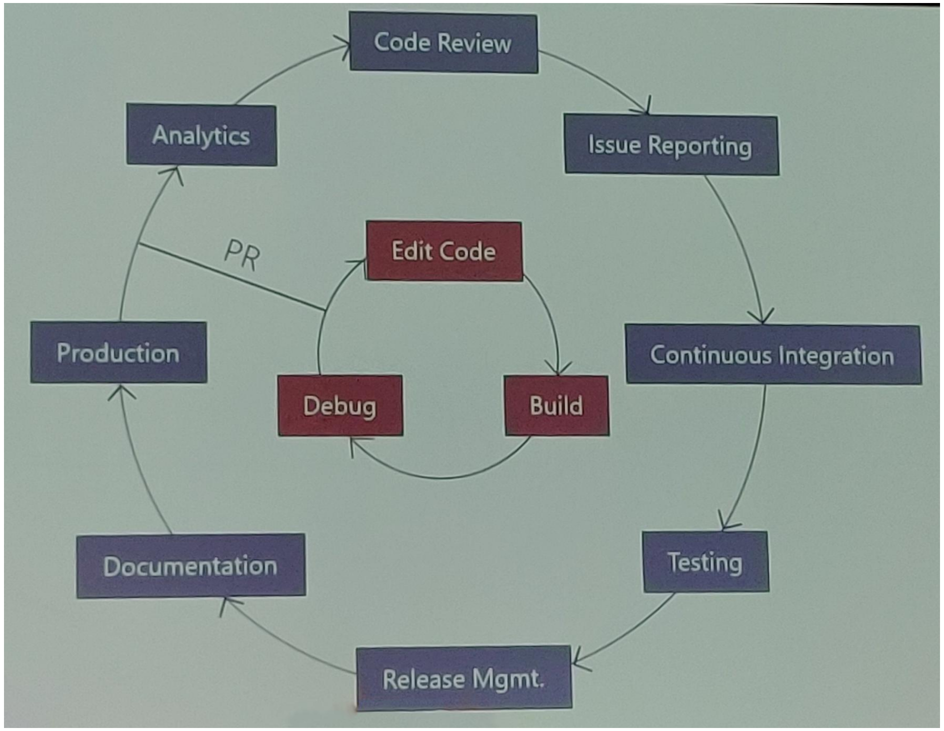
What is Software Engineering?

More than just writing code

The complete process of specifying, designing, developing, analyzing, deploying, and maintaining a software system.

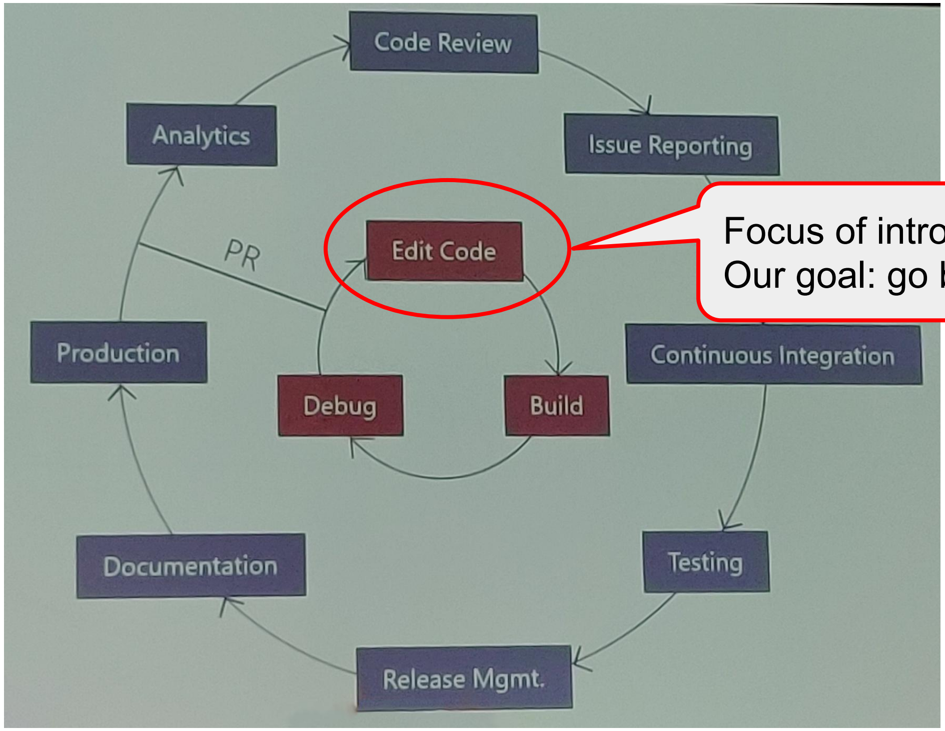
- Common Software Engineering tasks include:
  - Requirements engineering
  - Specification writing and documentation
  - Software architecture and design
  - Programming   Just one out of many important tasks!
  - Software testing and debugging
  - Maintenance and refactoring

The Role of Software Engineering in Practice



(Development workflow at Microsoft, Big Code summit 2019)

The Role of Software Engineering in Practice

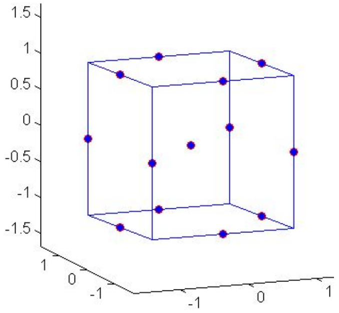


Focus of intro to programming courses.
Our goal: go beyond the inner cycle.

(Development workflow at Microsoft, Big Code summit 2019)

The Role of Software Engineering in Research

Experimental infrastructure is software, too!



Infrastructure

Design space exploration

1	0.34	0.81
2	0.52	0.32
3	0.21	0.53
4	0.81	0.22
...	...	...

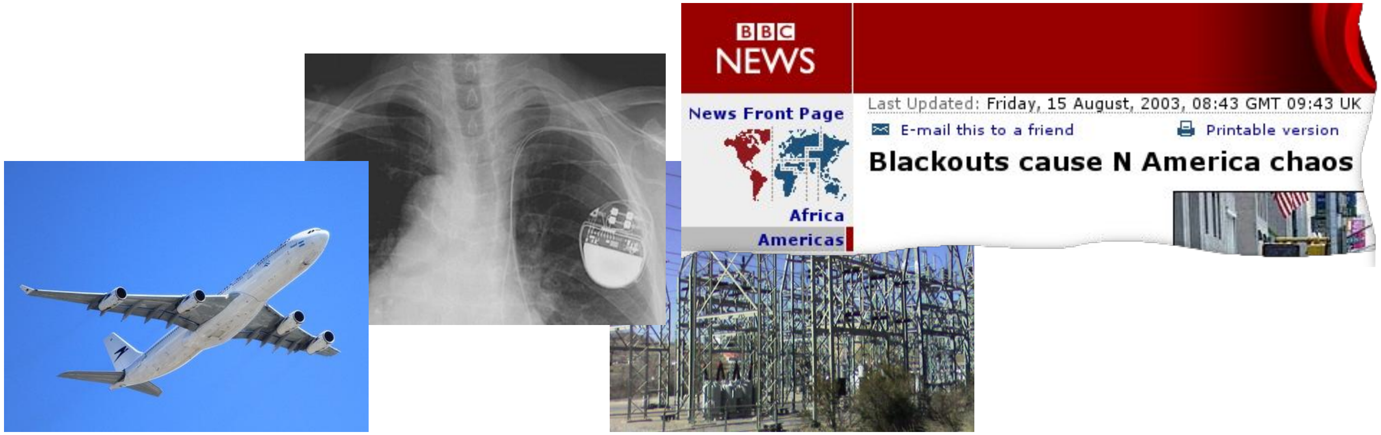
Example (automated debugging)

- 150 configurations, 1000+ benchmarks
- 1-85 hours per execution
- 200,000+ CPU hours (~23 CPU years)

Software bugs can lead to wrong scientific conclusions.

Why is Software Engineering important?

Software is eating the world!



Last Updated: Friday, 15 August, 2003, 08:43 GMT 09:43 UK

Blackouts cause N America chaos

c|net
Search CNET
Reviews
News
CNN
U.S. Edition
News Video TV Opinions More...
U.S. World Politics Tech Health Entertainment Living Travel
LATEX
R Git

Apple finally fixes 'gotofail' OS X

released a system software
e known as the "gotofail"

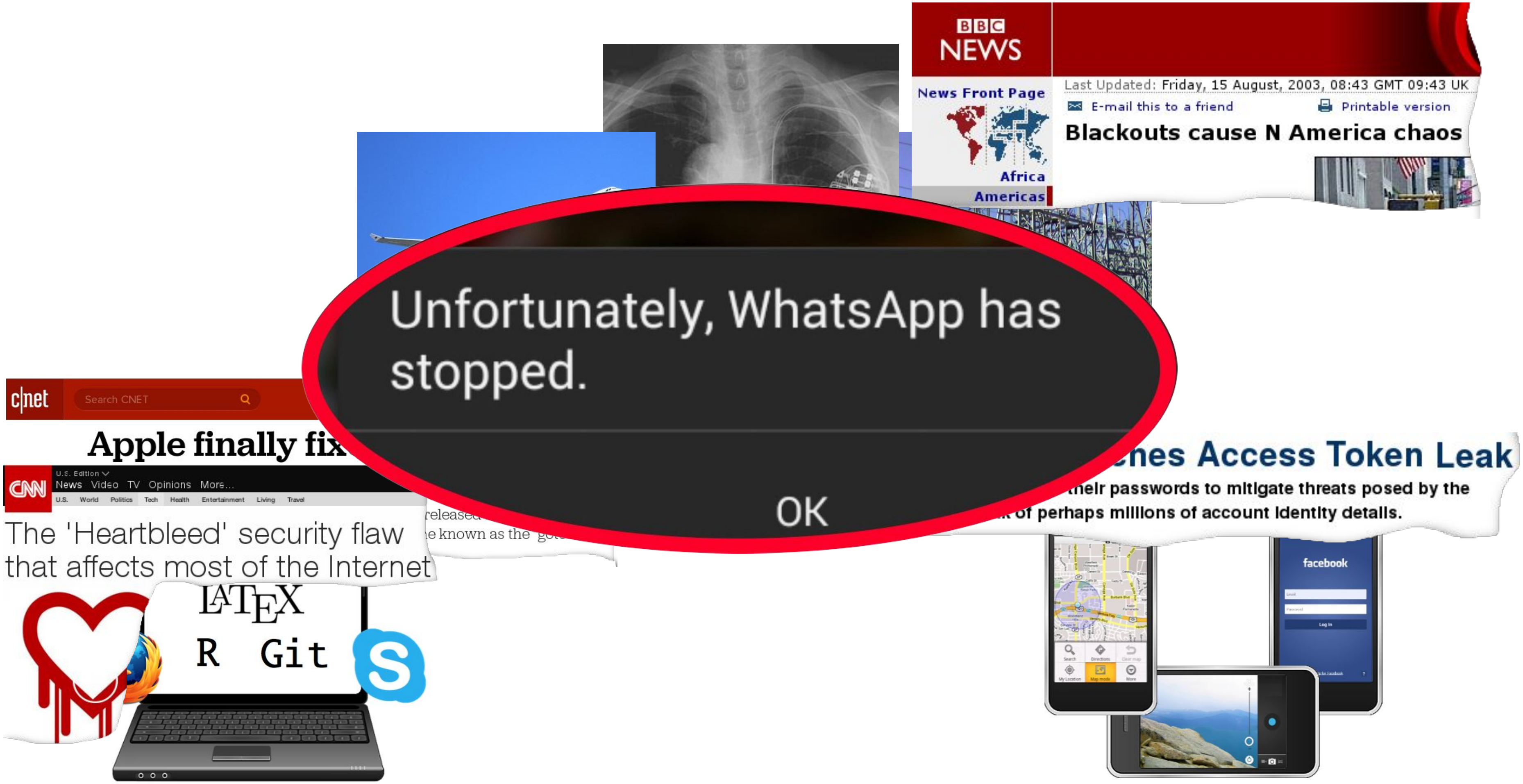
The 'Heartbleed' security flaw that affects most of the Internet

Facebook Patches Access Token Leak

Users should change their passwords to mitigate threats posed by the accidental leak of perhaps millions of account identity details.

Why is Software Engineering important?

Software is eating the world!



BBC NEWS

News Front Page

Africa

Americas

Last Updated: Friday, 15 August, 2003, 08:43 GMT 09:43 UK

E-mail this to a friend

Printable version

Blackouts cause N America chaos

Unfortunately, WhatsApp has stopped.

OK

c|net Search CNET

Apple finally fix

CNN U.S. Edition News Video TV Opinions More... U.S. World Politics Tech Health Entertainment Living Travel

The 'Heartbleed' security flaw that affects most of the Internet

released

e known as the go

hes Access Token Leak

heir passwords to mitigate threats posed by the

of perhaps millions of account identity details.

Why is Software Engineering important?

Software is complex!

- ~15 million lines of code

Let’s say 50 lines per page (0.05 mm)

- 300000 pages
- 15 m (49 ft)

Summary: Software Engineering

What is Software Engineering?

- The complete process of specifying, designing, developing, analyzing, and maintaining a software system.

Why is it important?

- Decomposes a complex engineering problem.
- Organizes processes and effort.
- Improves software reliability.
- Improves developer productivity.

Course overview: the big picture

- **Software processes, requirements, and specification**
  - Different software development processes.
  - Precise writing (requirements and specifications).
- **Software development**
  - Decompose a complex problem and build abstractions.
  - Improve your coding skills.
  - Effectively use version control (Git).
- **Software testing and debugging**
  - Write effective (unit) tests.
  - Hands-on experience, using testing and debugging techniques.
  - Continuous integration.
- **Class project**
  - Apply all of the above in a group project.

Expectations

- Programming experience and familiarity with one programming language (Java, C++, ...).
- Active participation in discussions.
- Teamwork and communication.
- Reflecting on and improving submitted assignments.

You must already know how to program.