Code Defenders

Crowdsourcing Effective Tests and Subtle Mutants with a Mutation Testing Game

José Miguel Rojas, Thomas White, Benjamin Clegg, Gordon Fraser

On Fault-revealing Tests

by Brian Hillegas (Flickr)

Manual Testing

Writing good tests is difficult and time-consuming.

EVOSUITE

Randoop

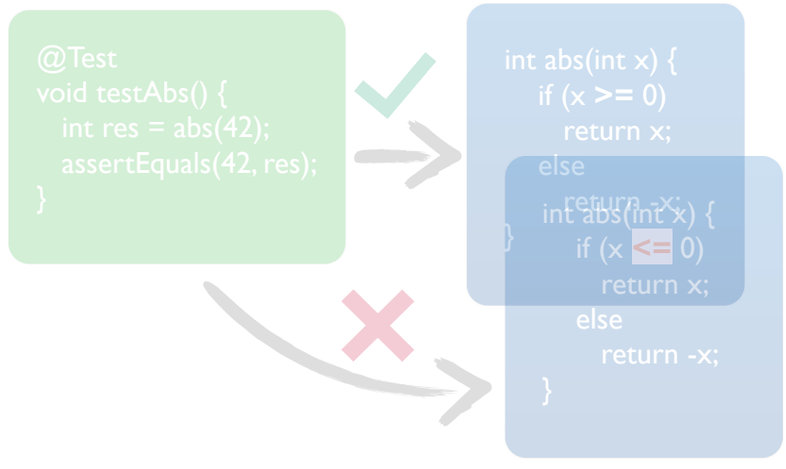
Automatic unit test generation

AgitarOne

Putting Java to the Test

Test Generation

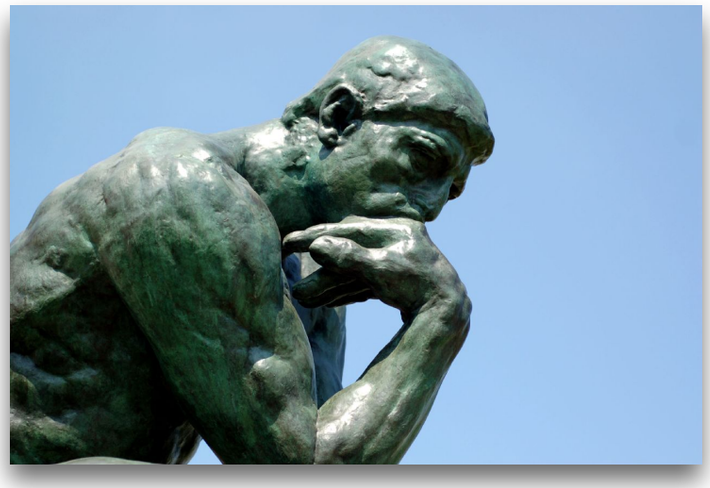
High coverage, but weak assertions.



Mutation Testing

Too many mutants, often trivial, redundant or equivalent.

On Fault-revealing Tests



by Brian Hillegas (Flickr)

Manual Testing

Writing good tests is difficult and time-consuming.

EVOSUITE

Randoop

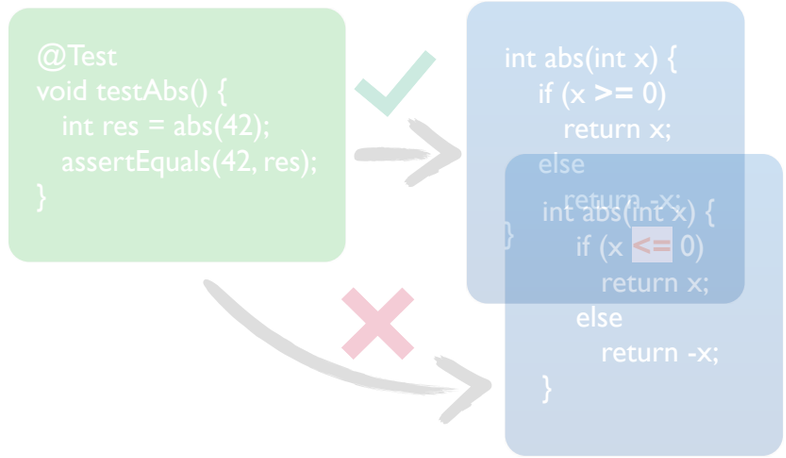
Automatic unit test generation f

AgitarOne

Putting Java to the Test

Test Generation

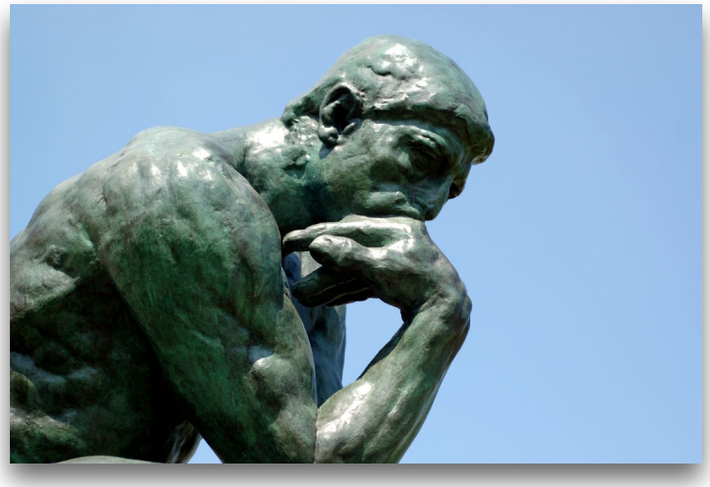
High coverage, but weak assertions.



Mutation Testing

Too many mutants, often trivial, redundant or equivalent.

On Fault-revealing Tests



by Brian Hillegas (Flickr)

Manual Testing

Writing good tests is difficult and time-consuming.

EVOSUITE

Randoop

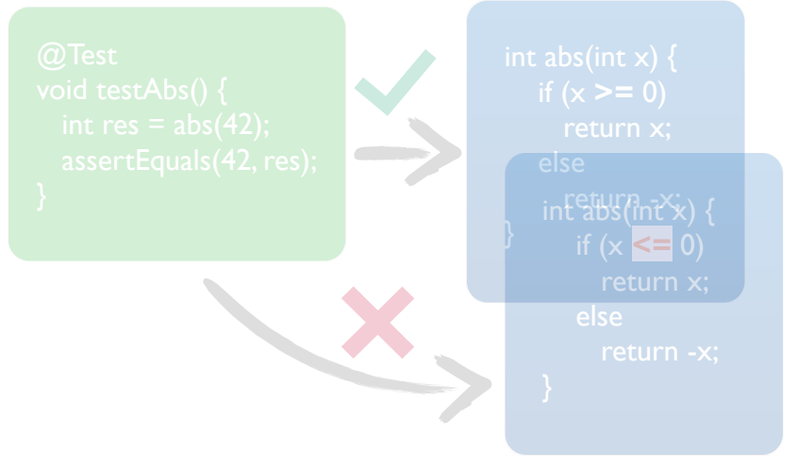
Automatic unit test generation f

AgitarOne

PUTTING JAVA TO THE TEST

Test Generation

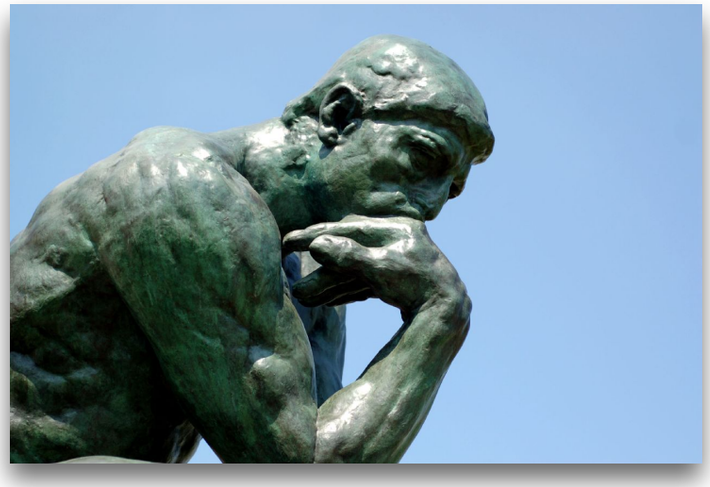
High coverage, but weak assertions.



Mutation Testing

Too many mutants, often trivial, redundant or equivalent.

On Fault-revealing Tests



by Brian Hillegas (Flickr)

Manual Testing

Writing good tests is difficult and time-consuming.

EVOSUITE

Randoop

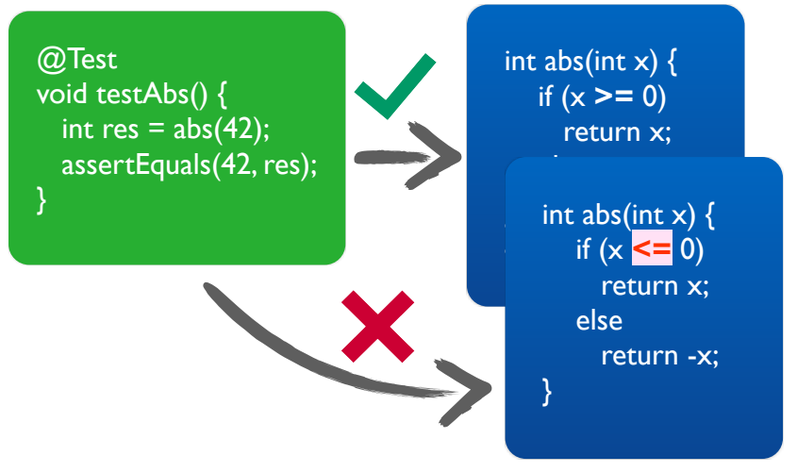
Automatic unit test generation f

AgitarOne

PUTTING JAVA TO THE TEST

Test Generation

High coverage, but weak assertions.



Mutation Testing

Too many mutants, often trivial, redundant or equivalent.

Code Defenders

by François Philipp (Flickr)

Software Testing Gamification

Supporting software testing education

by James Cridland (Flickr)

Crowdsourcing tests and mutants

Code Defenders

by François Philipp (Flickr)

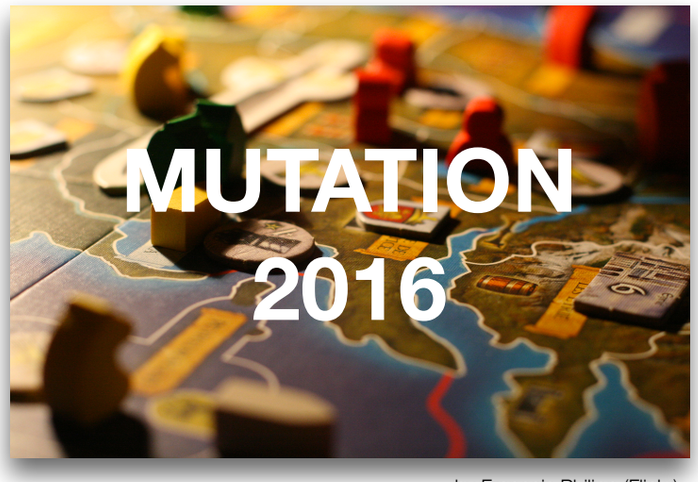
by James Cridland (Flickr)

Software Testing Gamification

Supporting software testing education

Crowdsourcing tests and mutants

Code Defenders



by François Philipp (Flickr)

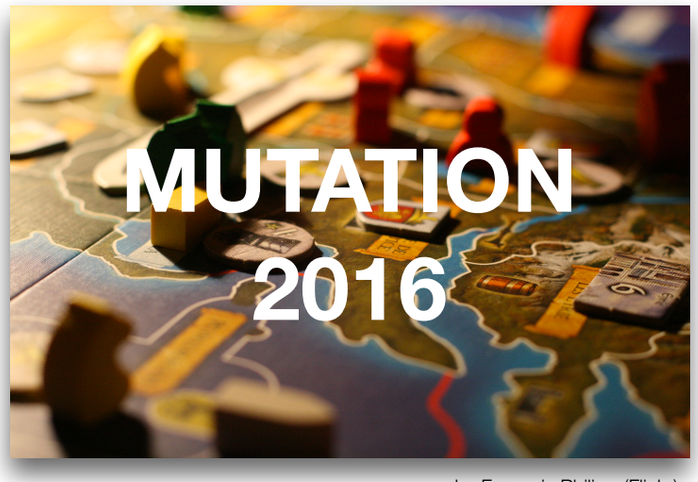
Software Testing Gamification

Supporting software testing education

by James Cridland (Flickr)

Crowdsourcing tests and mutants

Code Defenders



by François Philipp (Flickr)

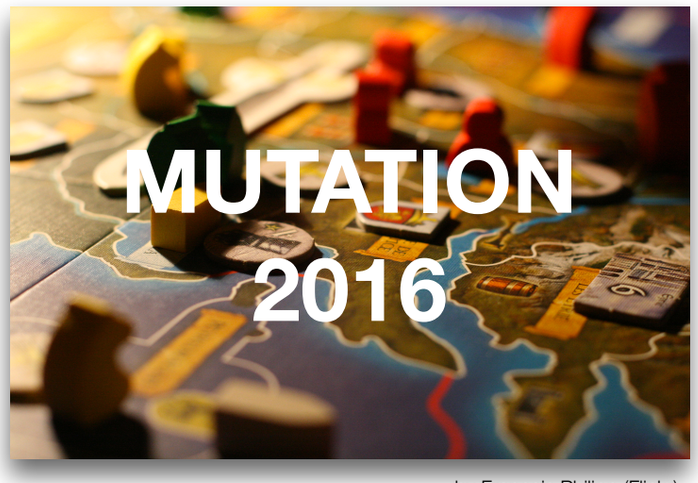
Software Testing Gamification

Supporting software testing education

by James Cridland (Flickr)

Crowdsourcing tests and mutants

Code Defenders



by François Philipp (Flickr)

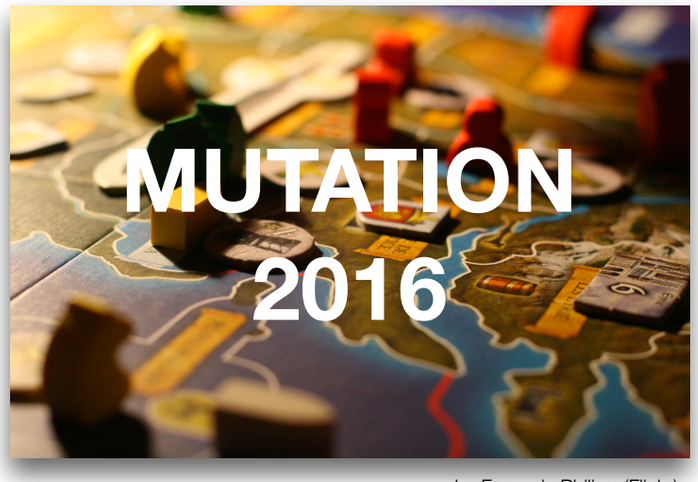
Software Testing
Gamification

Supporting software
testing education

by James Cridland (Flickr)

Crowdsourcing
tests and mutants

Code Defenders



by François Philipp (Flickr)

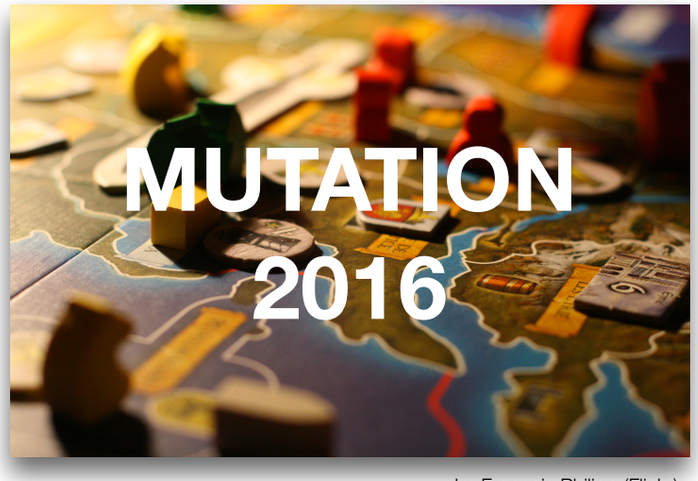
Software Testing Gamification

Supporting software testing education

by James Cridland (Flickr)

Crowdsourcing tests and mutants

Code Defenders



by François Philipp (Flickr)

Software Testing Gamification

Supporting software testing education

by James Cridland (Flickr)

Crowdsourcing tests and mutants

Code Defenders

Class Under Test

```
public class Arithmetics {
   public int abs(int x) {
     if (x >= 0)
         return x;
      else
         return -x;
   }
}
```

Code Defenders

Class Under Test

```
public class Arithmetics {
   public int abs(int x) {
     if (x >= 0)
         return x;
      else
         return -x;
   }
}
```

Attackers

Code Defenders

Class Under Test

```
public class Arithmetics {
   public int abs(int x) {
     if (x >= 0)
         return x;
      else
         return -x;
   }
}
```

Attackers

Defenders

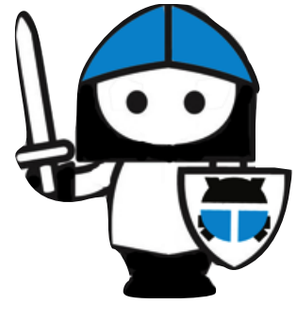
Code Defenders

Class Under Test

```java
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
          return x;
       else
          return -x;
   }
}
```

```java
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

Attackers



Defenders

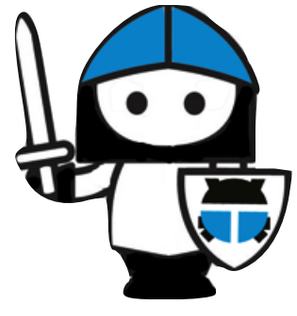
Code Defenders

Class Under Test

```java
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

```java
public class Arithmetics {
   public int abs(int x) {
     if (x >= 0)
         return x;
      else
         return -x;
   }
}
```

Attackers



Defenders

Code Defenders

Class Under Test

```java
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
          return x;
       else
          return -x;
   }
}
```

```java
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

Attackers

```java
public class TestArithmetics {
   @Test
   public void testAbs() {
      Arithmetics a;
      a = new Arithmetics();
      assertEquals(1, a.abs(-1));
   }
}
```

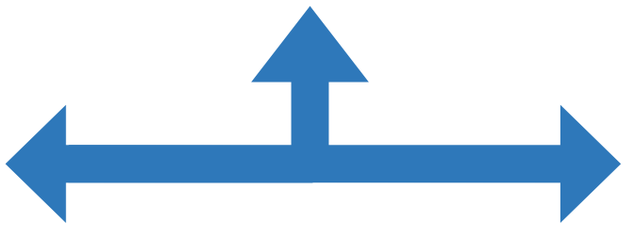
Defenders

Code Defenders

Class Under Test

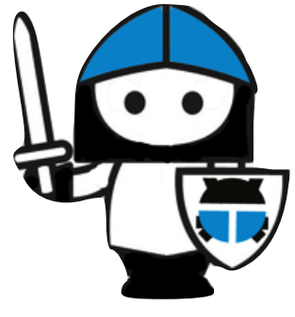
```java
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
          return x;
       else
          return -x;
   }
}
```

```java
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```



```java
public class TestArithmetics {
   @Test
   public void testAbs() {
      Arithmetics a;
      a = new Arithmetics();
      assertEquals(1, a.abs(-1));
   }
}
```

Attackers



Defenders

Code Defenders

Class Under Test

```
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
         return x;
      else
         return -x;
   }
}
```

Score points for surviving mutants

```
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

Attackers

```
public class TestArithmetics {
   @Test
   public void testAbs() {
      Arithmetics a;
      a = new Arithmetics();
      assertEquals(1, a.abs(-1));
   }
}
```

Defenders

Code Defenders

Class Under Test

```
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
         return x;
      else
         return -x;
   }
}
```

Score points for surviving mutants

```
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

Attackers

Score points for effective tests

```
public class TestArithmetics {
   @Test
   public void testAbs() {
      Arithmetics a;
      a = new Arithmetics();
      assertEquals(1, a.abs(-1));
   }
}
```

Defenders

Code Defenders

Class Under Test

```
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
         return x;
      else
         return -x;
   }
}
```

Score points for surviving mutants

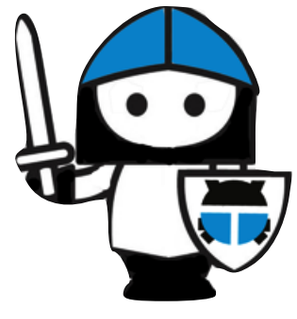
```
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

Score points for effective tests

```
public class TestArithmetics {
   @Test
   public void testAbs() {
      Arithmetics a;
      a = new Arithmetics();
      assertEquals(1, a.abs(-1));
   }
}
```

Attackers

Equivalent Mutant Duels



Defenders

Code Defenders

Class Under Test

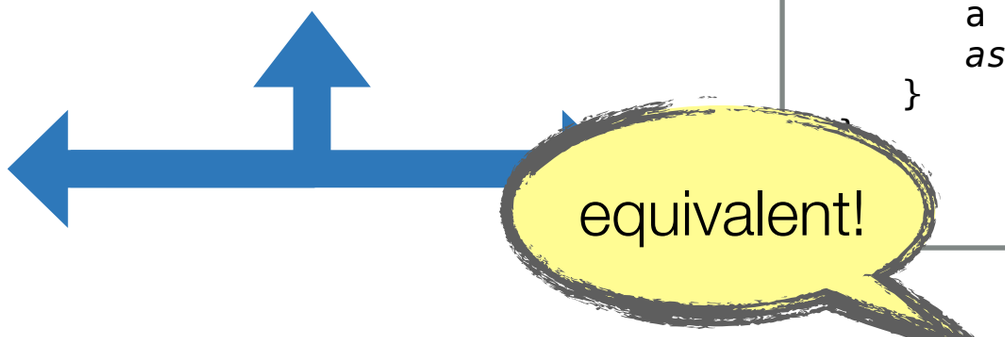
```
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
         return x;
      else
         return -x;
   }
}
```

Score points for surviving mutants

```
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

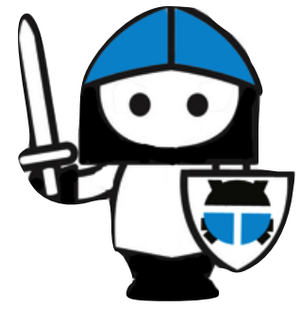
Score points for effective tests

```
public class TestArithmetics {
   @Test
   public void testAbs() {
      Arithmetics a;
      a = new Arithmetics();
      assertEquals(1, a.abs(-1));
   }
```



Attackers

Equivalent Mutant Duels



Defenders

Code Defenders

Class Under Test

```
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
          return x;
       else
          return -x;
   }
}
```

Score points for surviving mutants

```
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

Score points for effective tests

```
public class TestArithmetics {
   @Test
   public void testAbs() {
      Arithmetics a;
      a = new Arithmetics();
      assertEquals(1, a.abs(-1));
   }
```

no way! here is a killing test!

equivalent!

Attackers

Equivalent Mutant Duels

Defenders

Code Defenders

Class Under Test

```
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
          return x;
       else
          return -x;
   }
}
```

Score points for surviving mutants

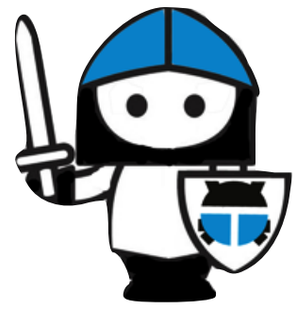
```
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

Score points for effective tests

```
public class TestArithmetics {
   @Test
   public void testAbs() {
      Arithmetics a;
      a = new Arithmetics();
      assertEquals(1, a.abs(-1));
   }
```

no way! here is a killing test!

equivalent!



Attackers

Equivalent Mutant Duels

Defenders

Code Defenders

Class Under Test

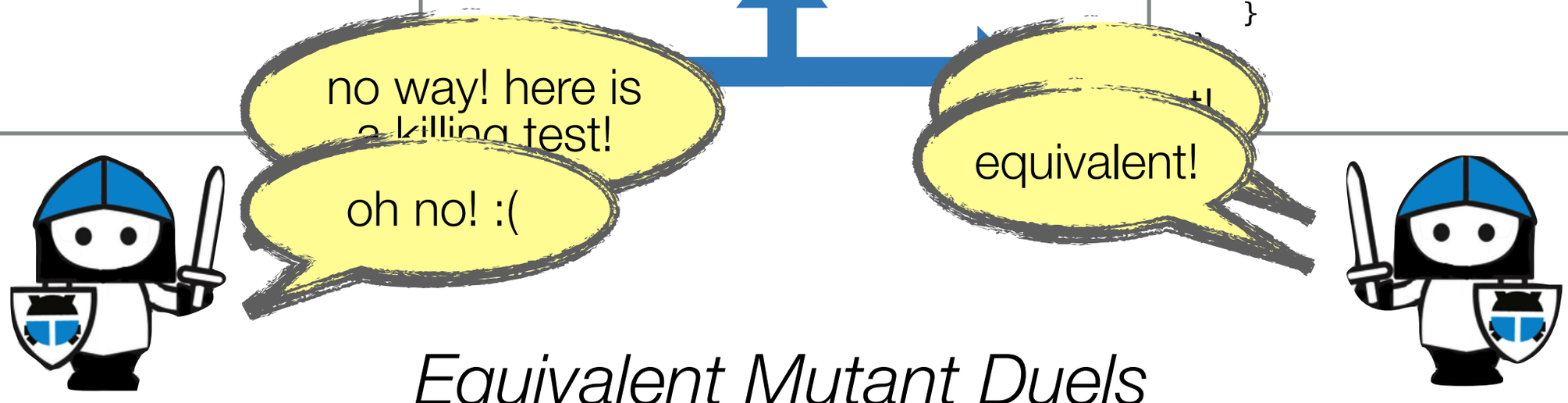
```
public class Arithmetics {
   public int abs(int x) {
      if (x >= 0)
         return x;
      else
         return -x;
   }
}
```

Score points for surviving mutants

```
public class Arithmetics {
   public int abs(int x) {
      if (x < 0)
         return x;
      else
         return -x;
   }
}
```

Score points for effective tests

```
public class TestArithmetics {
   @Test
   public void testAbs() {
      Arithmetics a;
      a = new Arithmetics();
      assertEquals(1, a.abs(-1));
   }
```

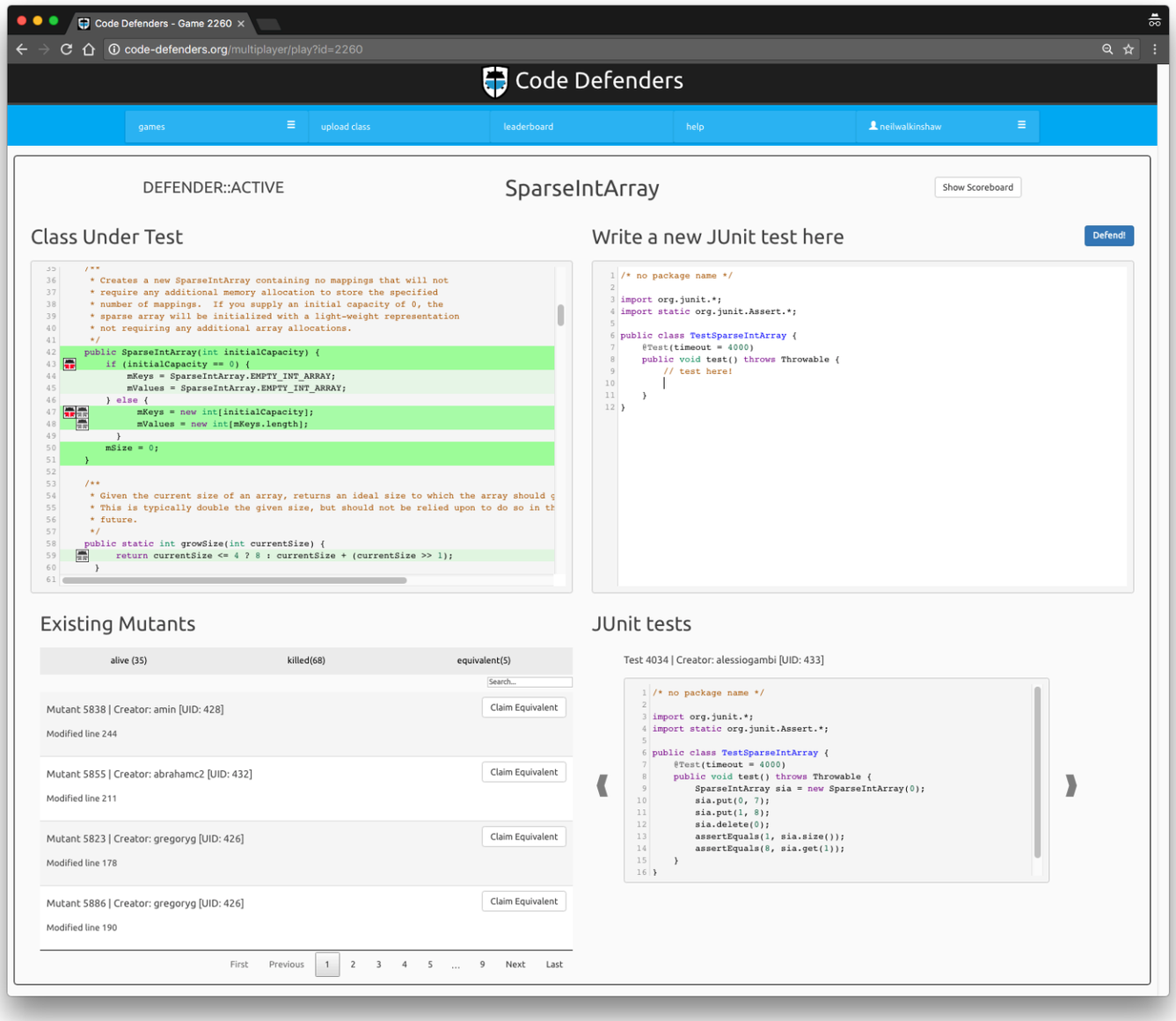


Equivalent Mutant Duels

Attackers

Defenders

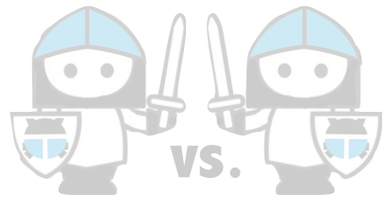
Code Defenders



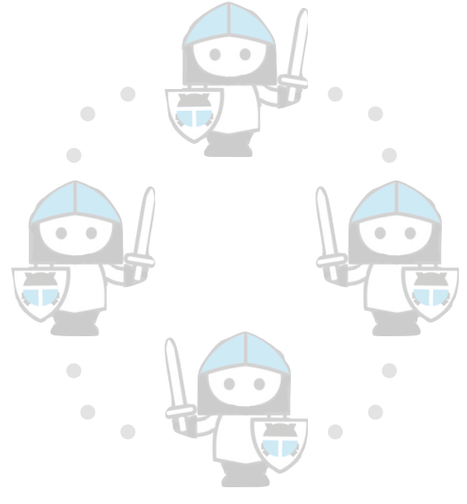
Single-player

vs.

EVOSUITE / MAJOR

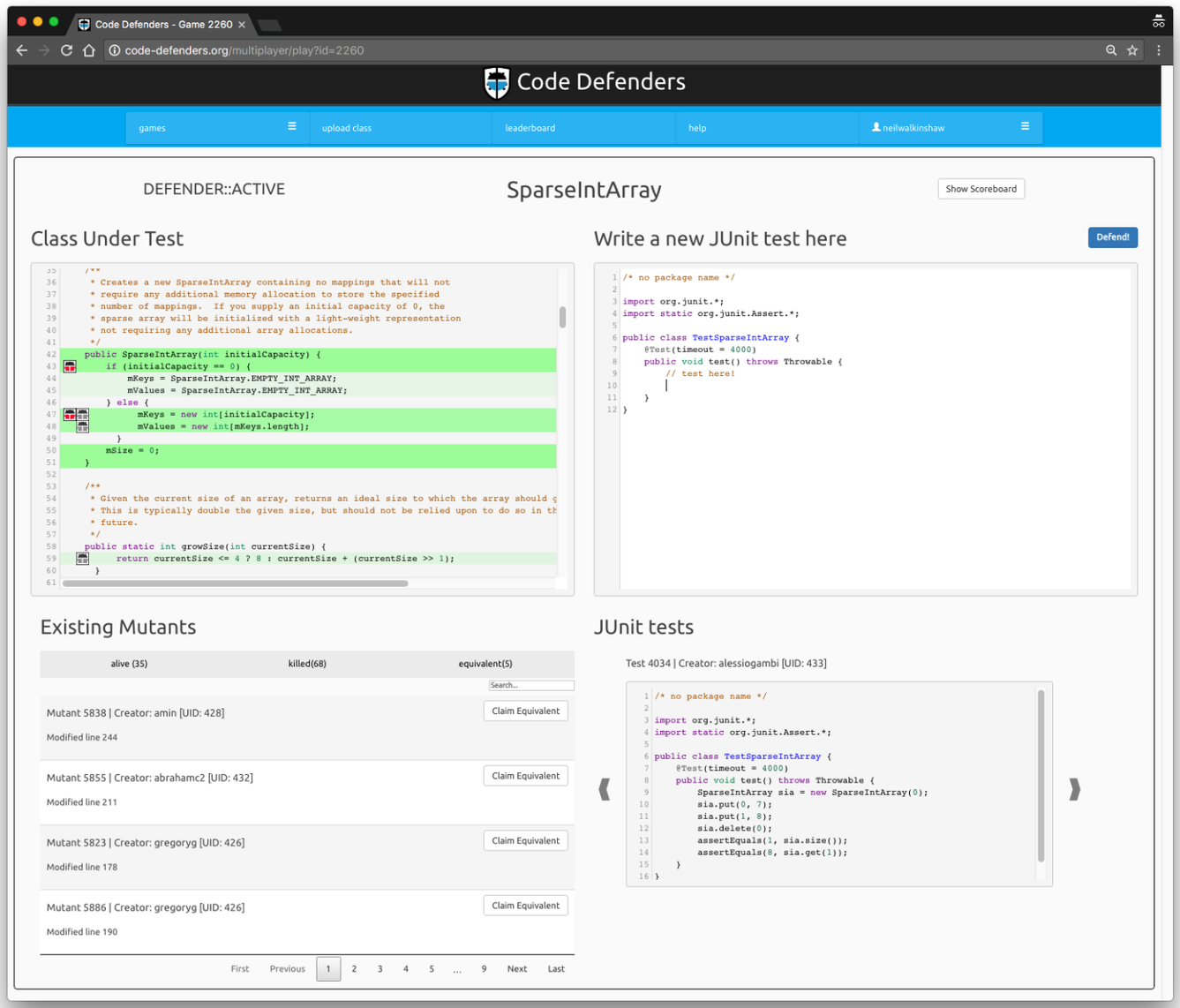


Two-player



Multi-player

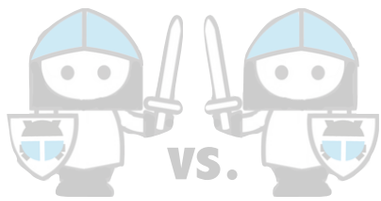
Code Defenders



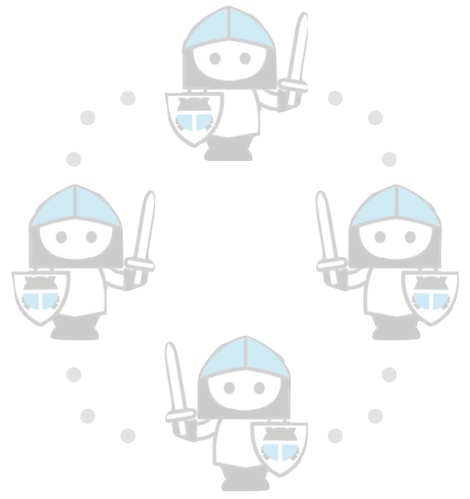
Single-player

vs.

EVOSUITE / MAJOR

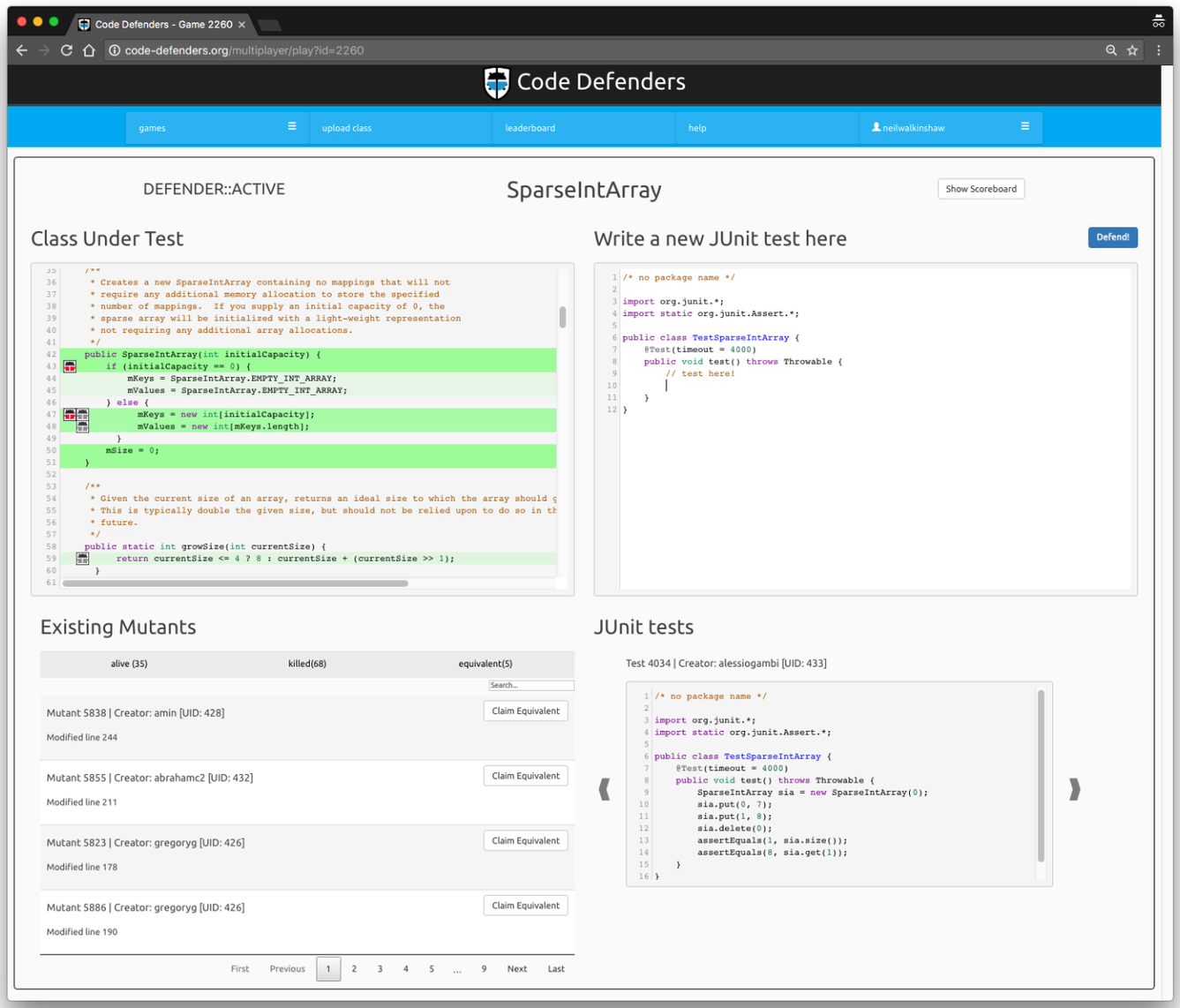


Two-player



Multi-player

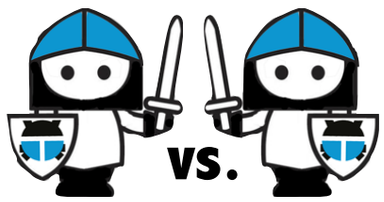
Code Defenders



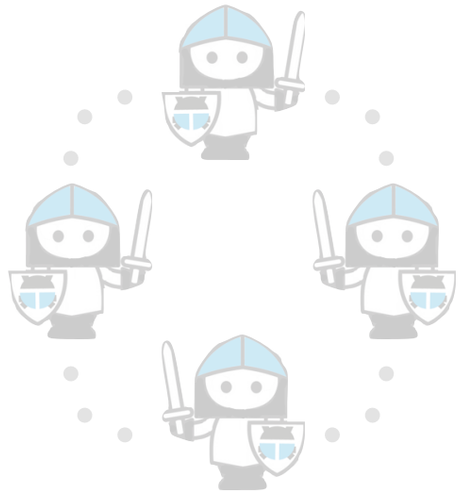
Single-player

vs.

EVOSUITE / MAJOR

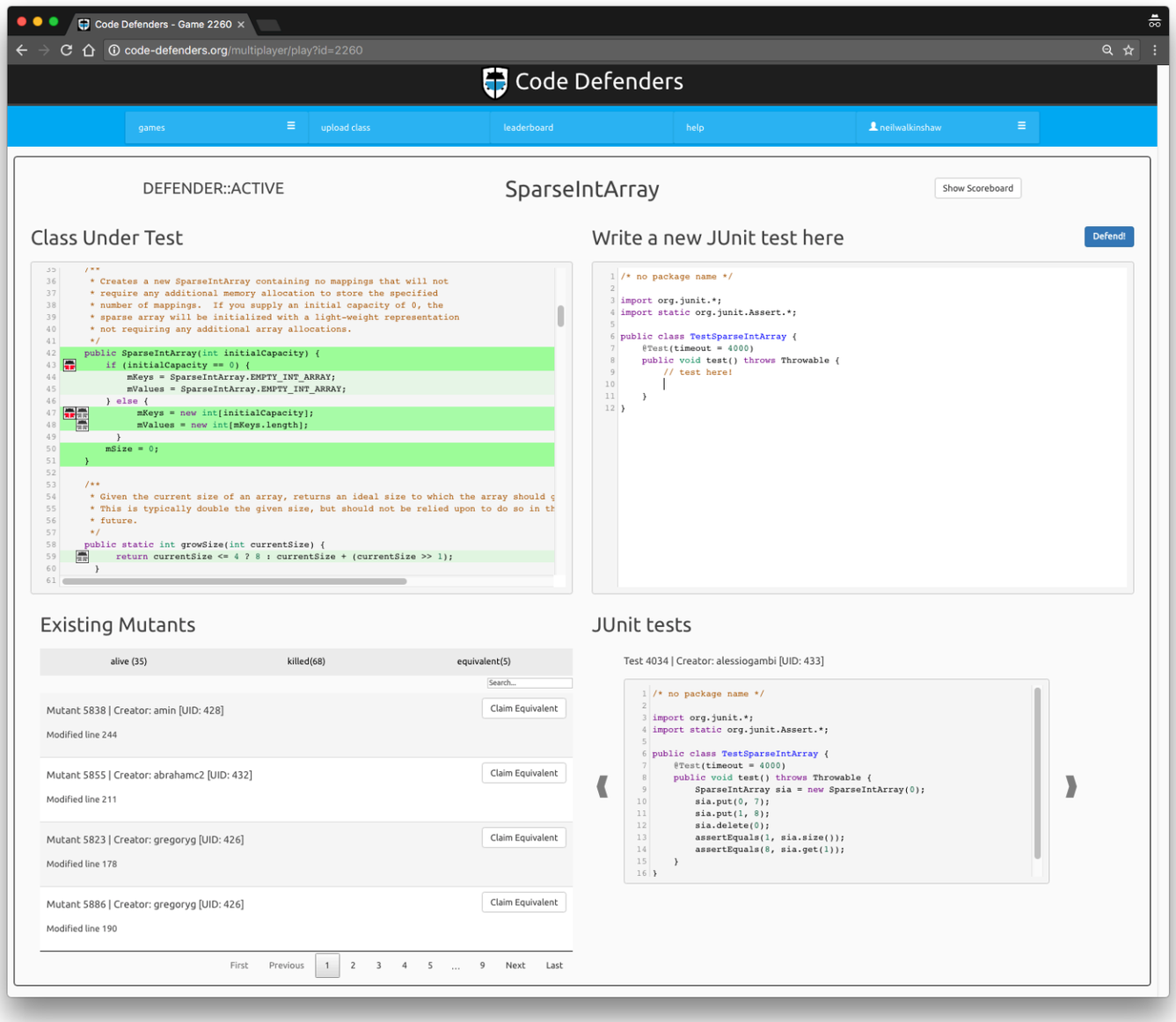


Two-player



Multi-player

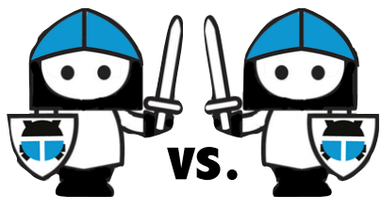
Code Defenders



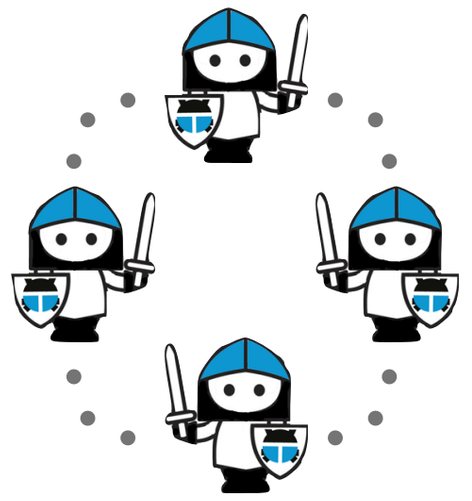
Single-player

vs.

EVOSUITE / MAJOR



Two-player



Multi-player

Code Defenders

DEFENDER::ACTIVE

SparseIntArray

Class Under Test

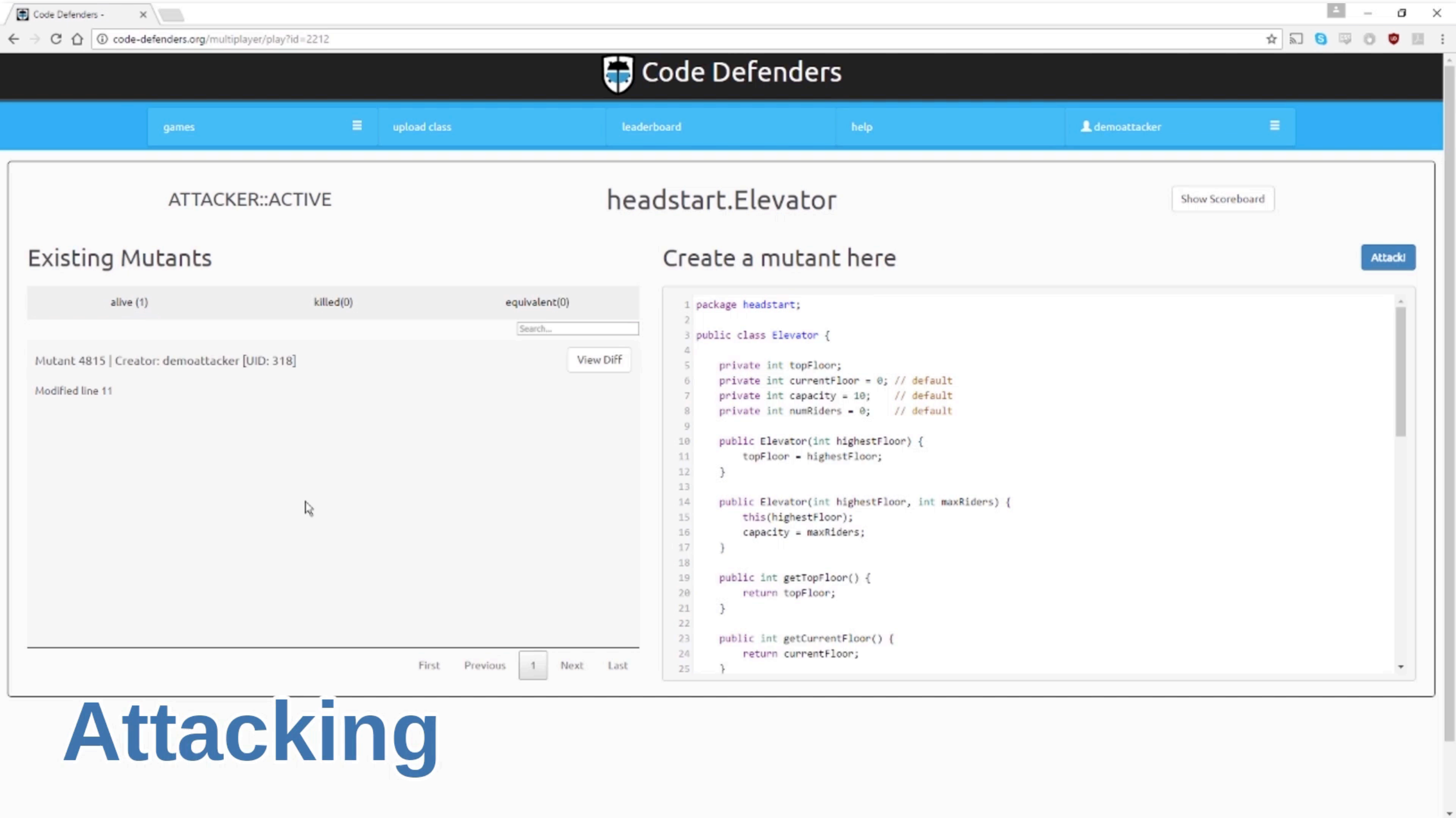
```
35      /**
36       * Creates a new SparseIntArray containing no mappings that will not
37       * require any additional memory allocation to store the specified
38       * number of mappings.  If you supply an initial capacity of 0, the
39       * sparse array will be initialized with a light-weight representation
40       * not requiring any additional array allocations.
41       */
42      public SparseIntArray(int initialCapacity) {
43          if (initialCapacity == 0) {
44              mKeys = SparseIntArray.EMPTY_INT_ARRAY;
45              mValues = SparseIntArray.EMPTY_INT_ARRAY;
46          } else {
47              mKeys = new int[initialCapacity];
48              mValues = new int[mKeys.length];
49          }
50          mSize = 0;
51      }
52
53      /**
54       * Given the current size of an array, returns an ideal size to which the array should g
55       * This is typically double the given size, but should not be relied upon to do so in th
56       * future.
57       */
58      public static int growSize(int currentSize) {
59          return currentSize <= 4 ? 8 : currentSize + (currentSize >> 1);
60      }
61
```

Write a new JUnit test here

```
1  /* no package name */
2
3  import org.junit.*;
4  import static org.junit.Assert.*;
5
6  public class TestSparseIntArray {
7      @Test(timeout = 4000)
8      public void test() throws Throwable {
9          // test here!
10
11     }
12 }
```

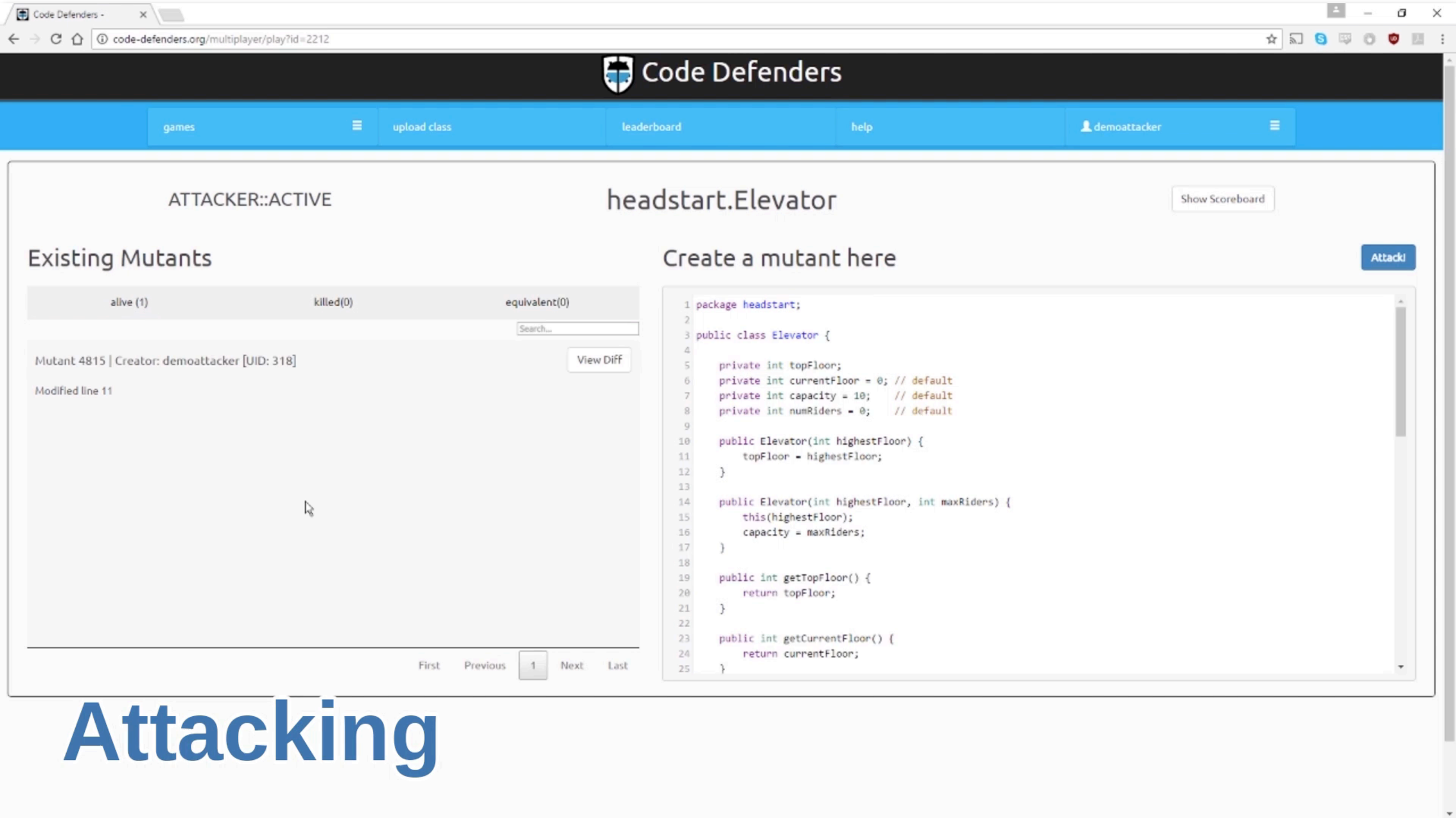
Existing Mutants

JUnit tests



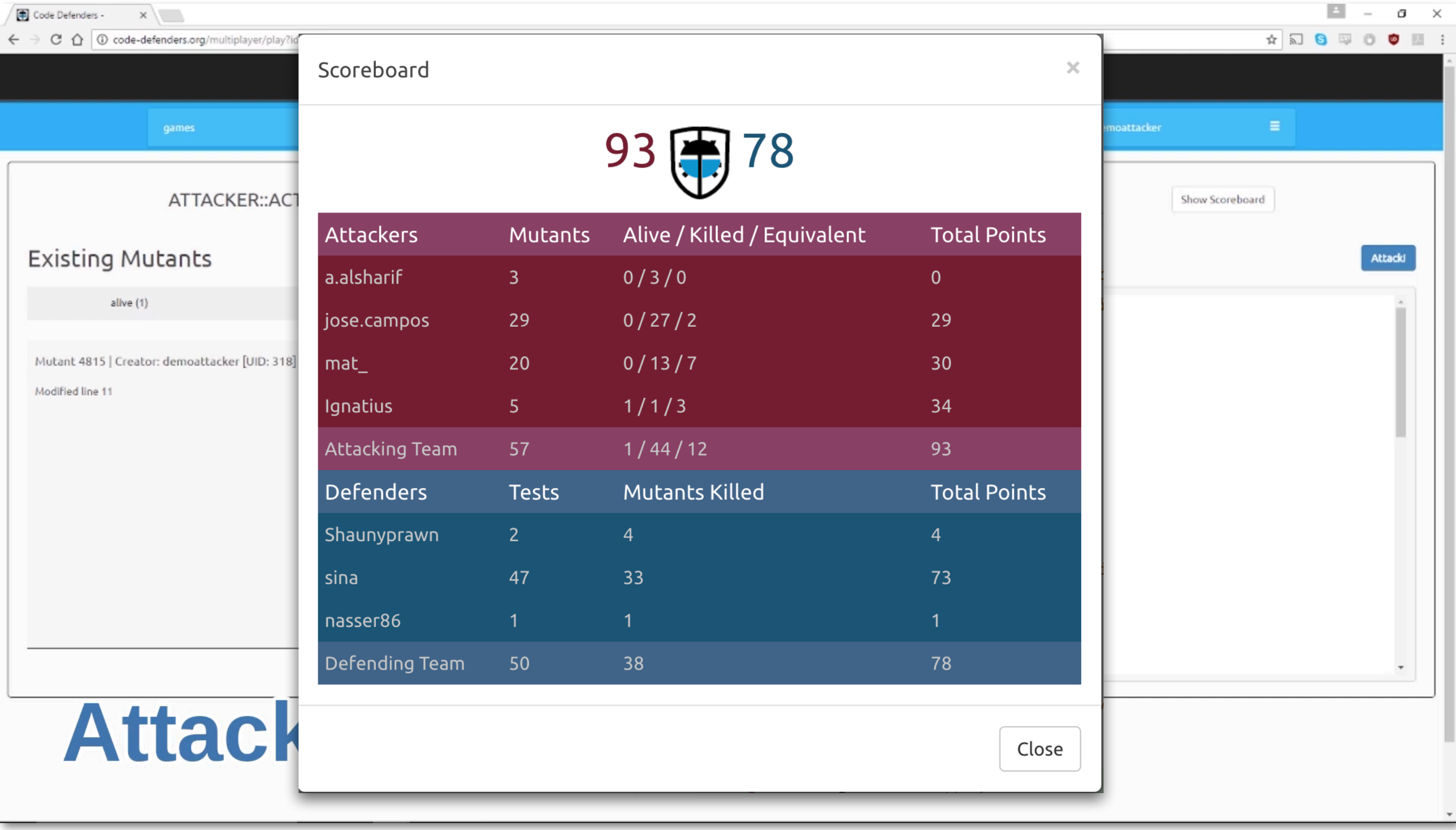
Attacking

ICST'17 Best Tool Demo Award



Attacking

ICST'17 Best Tool Demo Award



Attackers	Mutants	Alive / Killed / Equivalent	Total Points
a.alsharif	3	0 / 3 / 0	0
jose.campos	29	0 / 27 / 2	29
mat_	20	0 / 13 / 7	30
Ignatius	5	1 / 1 / 3	34
Attacking Team	57	1 / 44 / 12	93
Defenders	**Tests**	**Mutants Killed**	**Total Points**
Shaunyprawn	2	4	4
sina	47	33	73
nasser86	1	1	1
Defending Team	50	38	78

ICST'17 Best Tool Demo Award

Does Gamification Work?

Does Gamification Work?

Does Crowdsourcing Work?

Does Gamification Work?

Controlled Study

Does Code Defenders lead to better tests than regular unit testing? Is it fun to play?

Does Gamification Work?

Controlled Study

Does Code Defenders lead to better tests than regular unit testing? Is it fun to play?

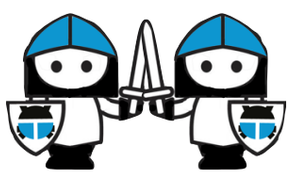
41

Does Gamification Work?

Controlled Study

Does Code Defenders lead to better tests than regular unit testing? Is it fun to play?

41



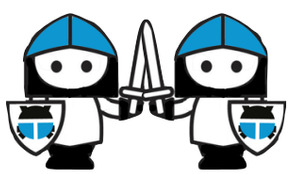
2-player

Does Gamification Work?

Controlled Study

Does Code Defenders lead to better tests than regular unit testing? Is it fun to play?

41



2-player

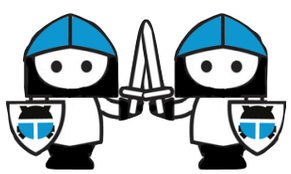
1

Does Gamification Work?

Controlled Study

Does Code Defenders lead to better tests than regular unit testing? Is it fun to play?

41



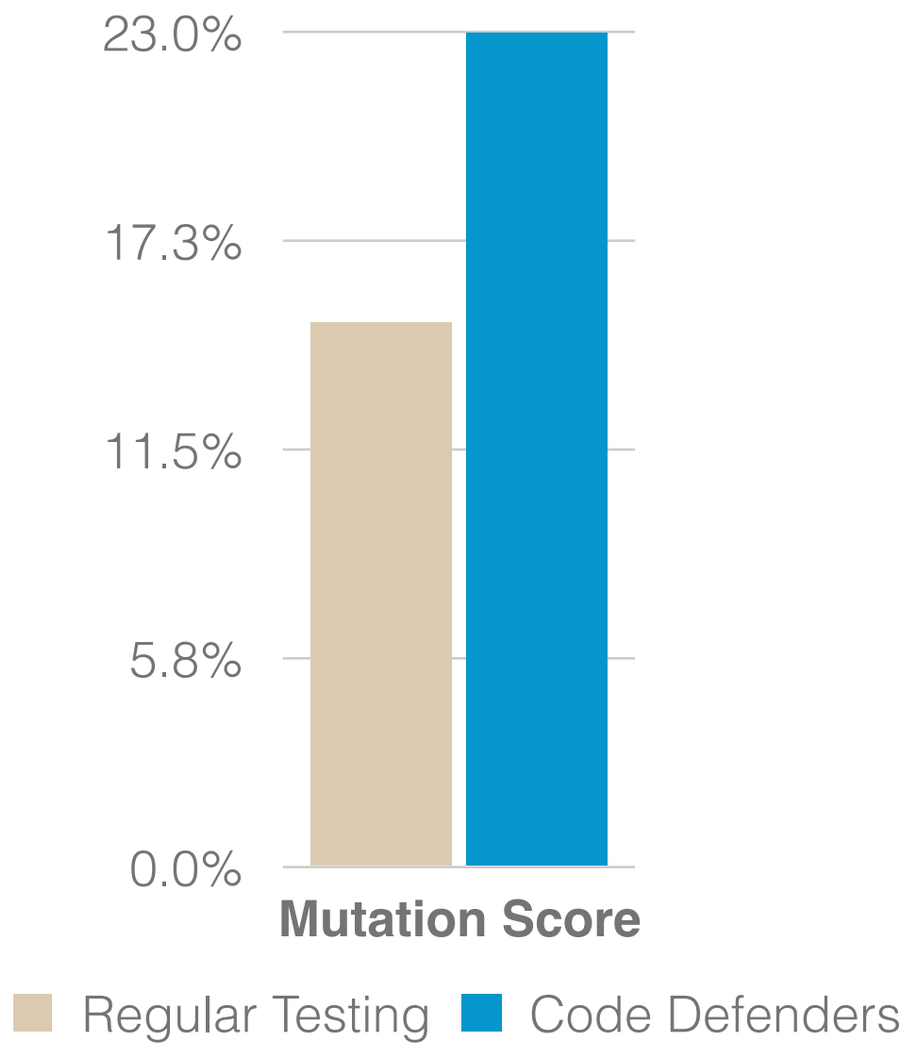
2-player

1

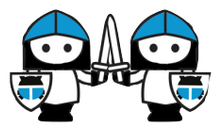
2

Does Gamification Work?

Does Gamification Work?



41

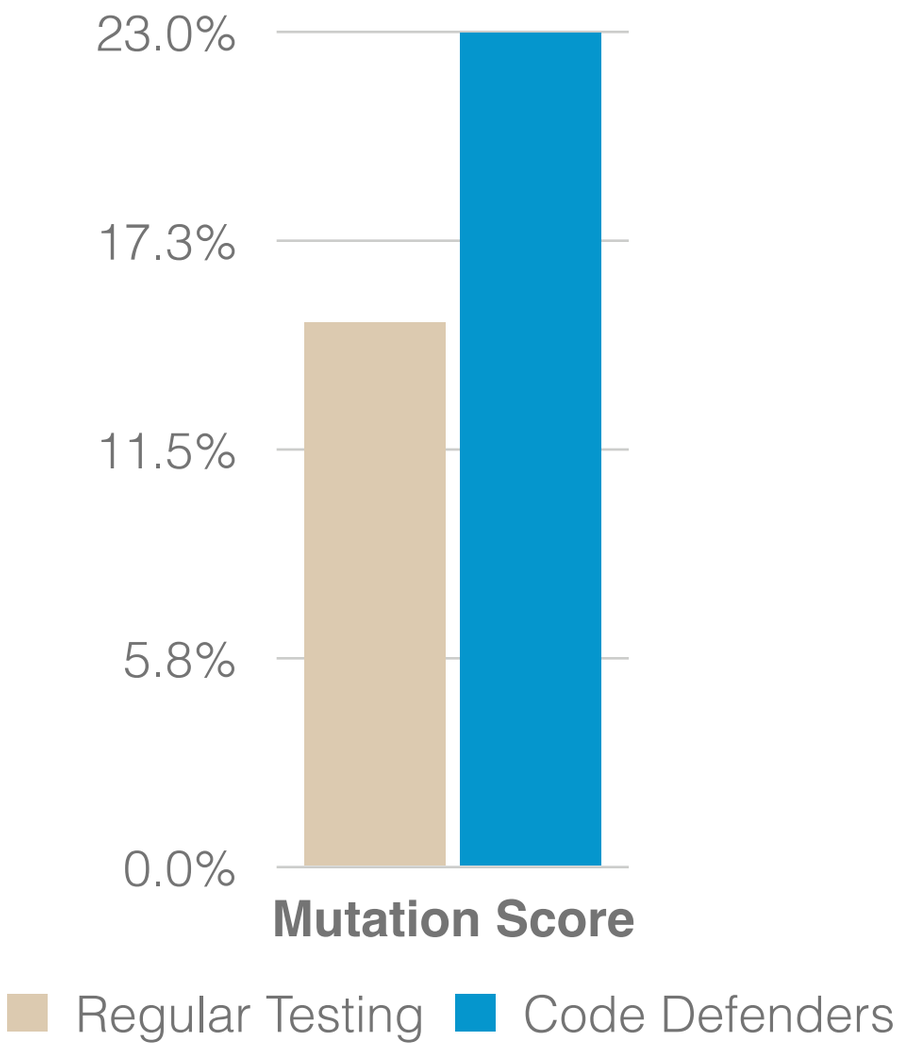


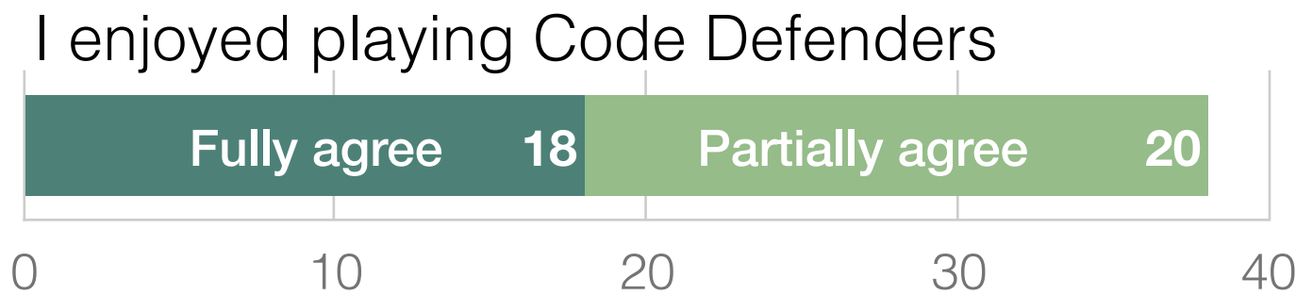
2-player

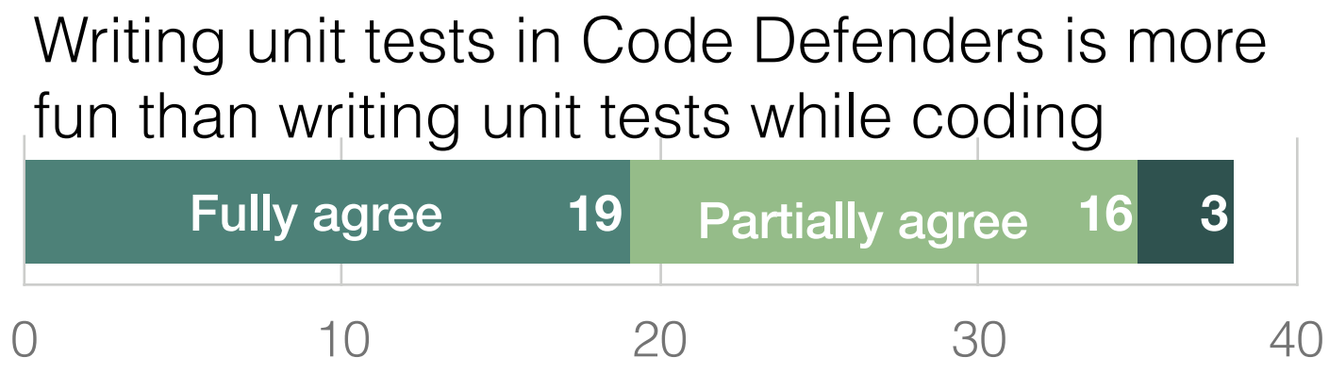
1

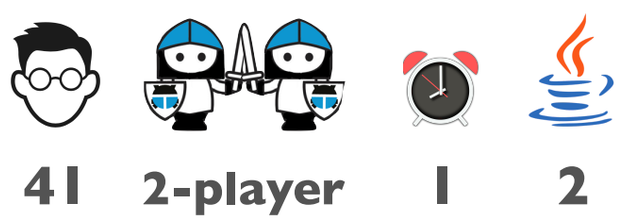
2

Does Gamification Work?

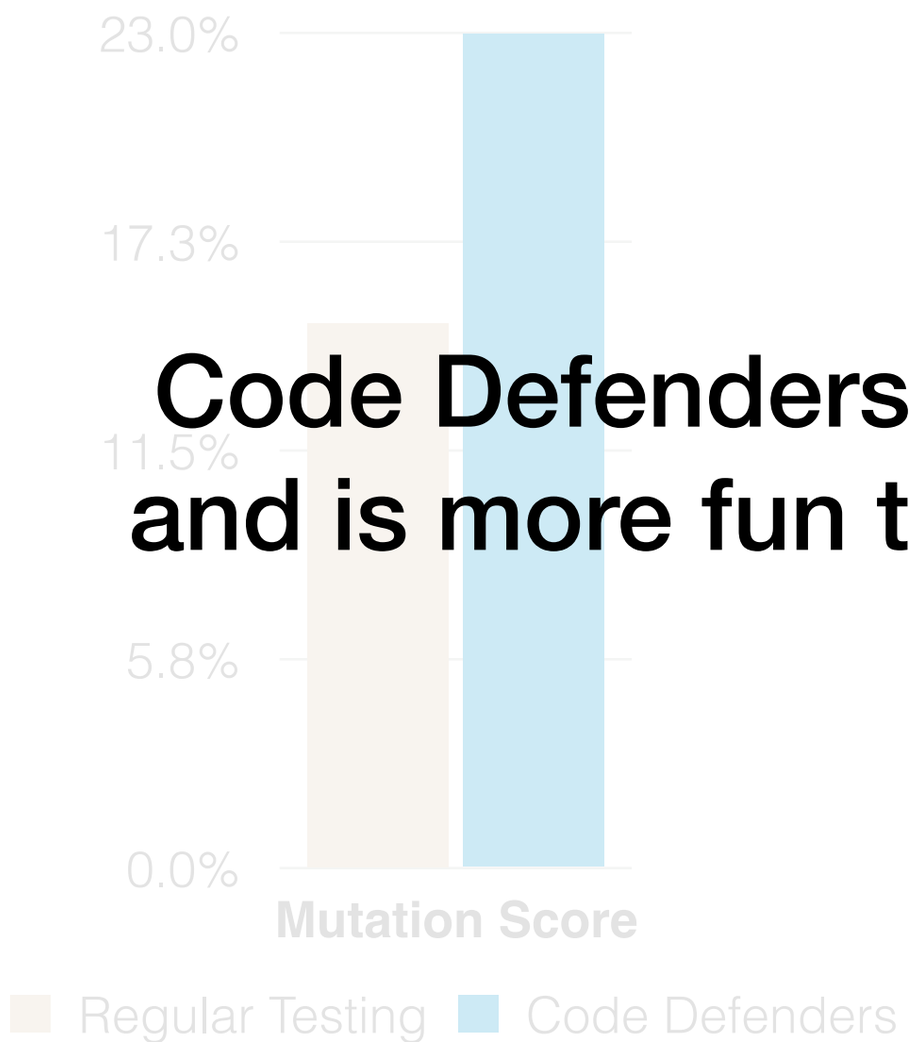


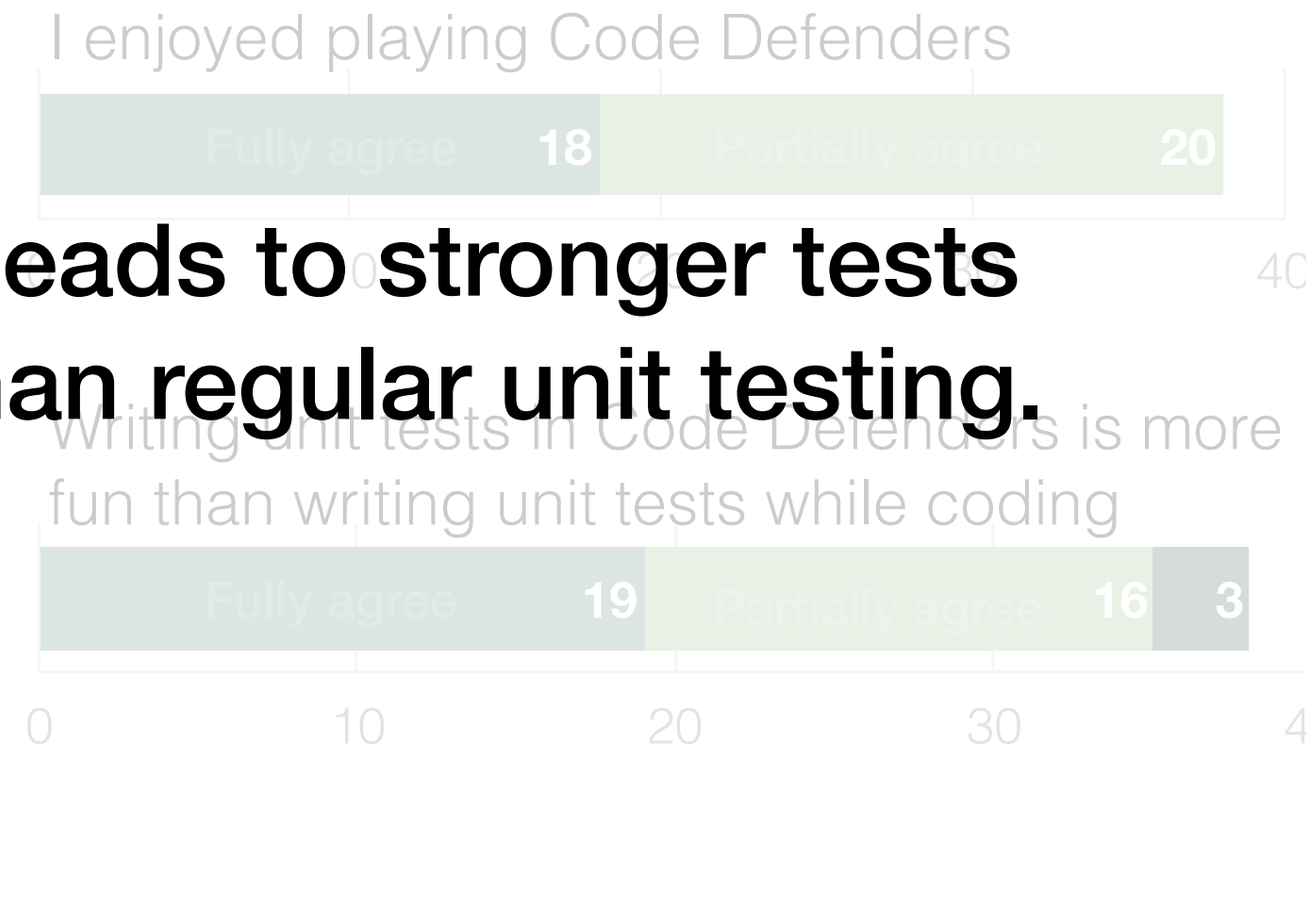






Does Gamification Work?





Code Defenders leads to stronger tests and is more fun than regular unit testing.

Does Crowdsourcing Work?

Online Study

Does using Code Defenders in a crowdsourcing scenario lead to better tests and mutants than those generated by test generation and mutation testing tools?

Does Crowdsourcing Work?

Online Study

Does using Code Defenders in a crowdsourcing scenario lead to better tests and mutants than those generated by test generation and mutation testing tools?

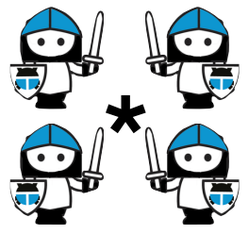
35

Does Crowdsourcing Work?

Online Study

Does using Code Defenders in a crowdsourcing scenario lead to better tests and mutants than those generated by test generation and mutation testing tools?

35



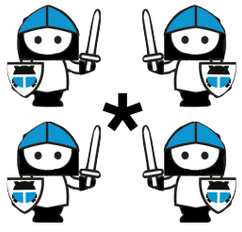
6-10

Does Crowdsourcing Work?

Online Study

Does using Code Defenders in a crowdsourcing scenario lead to better tests and mutants than those generated by test generation and mutation testing tools?

35



6-10

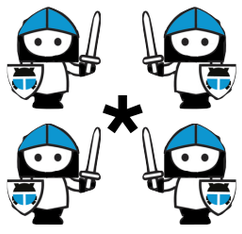
24

Does Crowdsourcing Work?

Online Study

Does using Code Defenders in a crowdsourcing scenario lead to better tests and mutants than those generated by test generation and mutation testing tools?

35

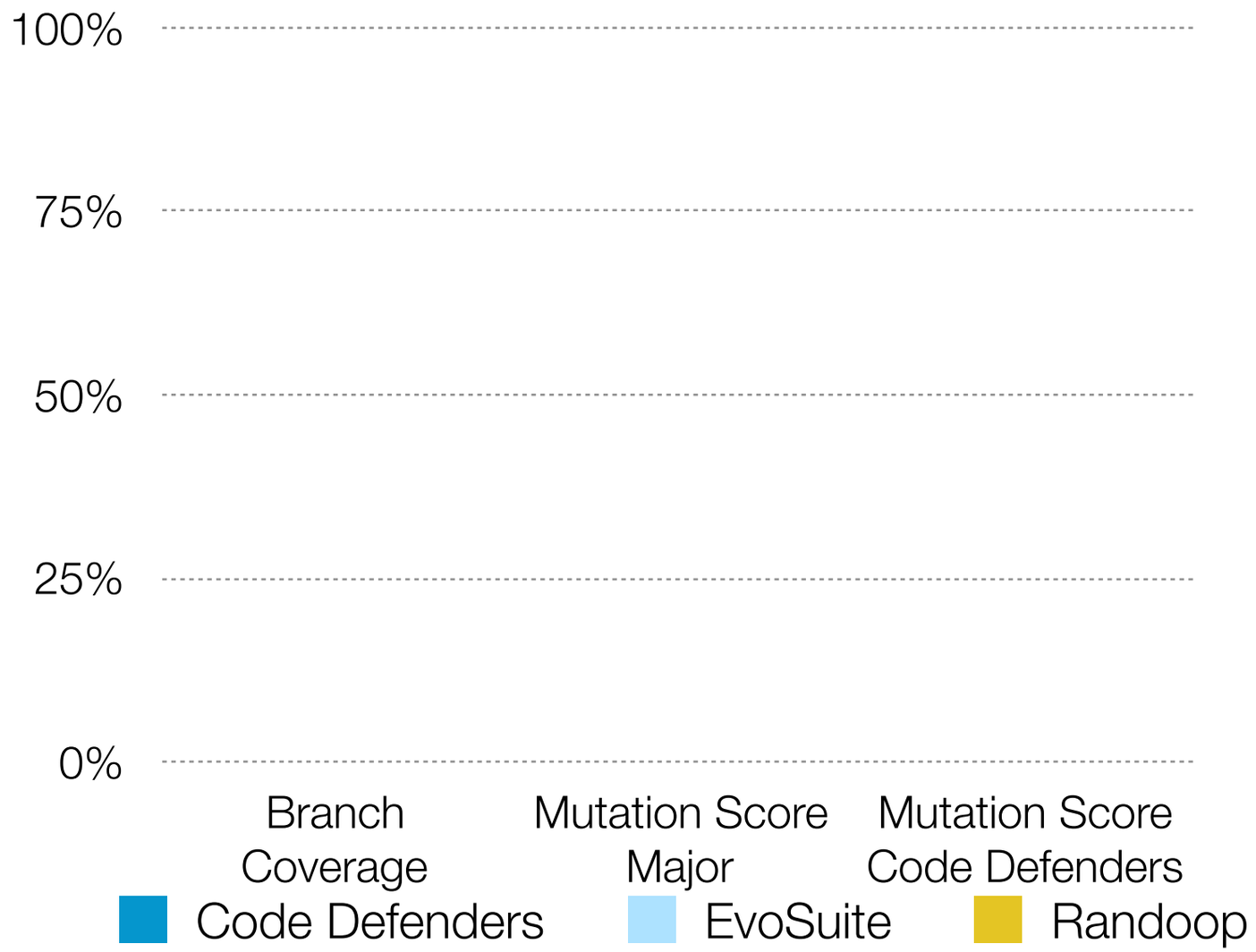


6-10

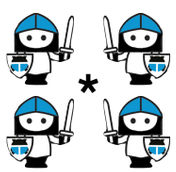
24

20

Does Crowdsourcing Work?



35

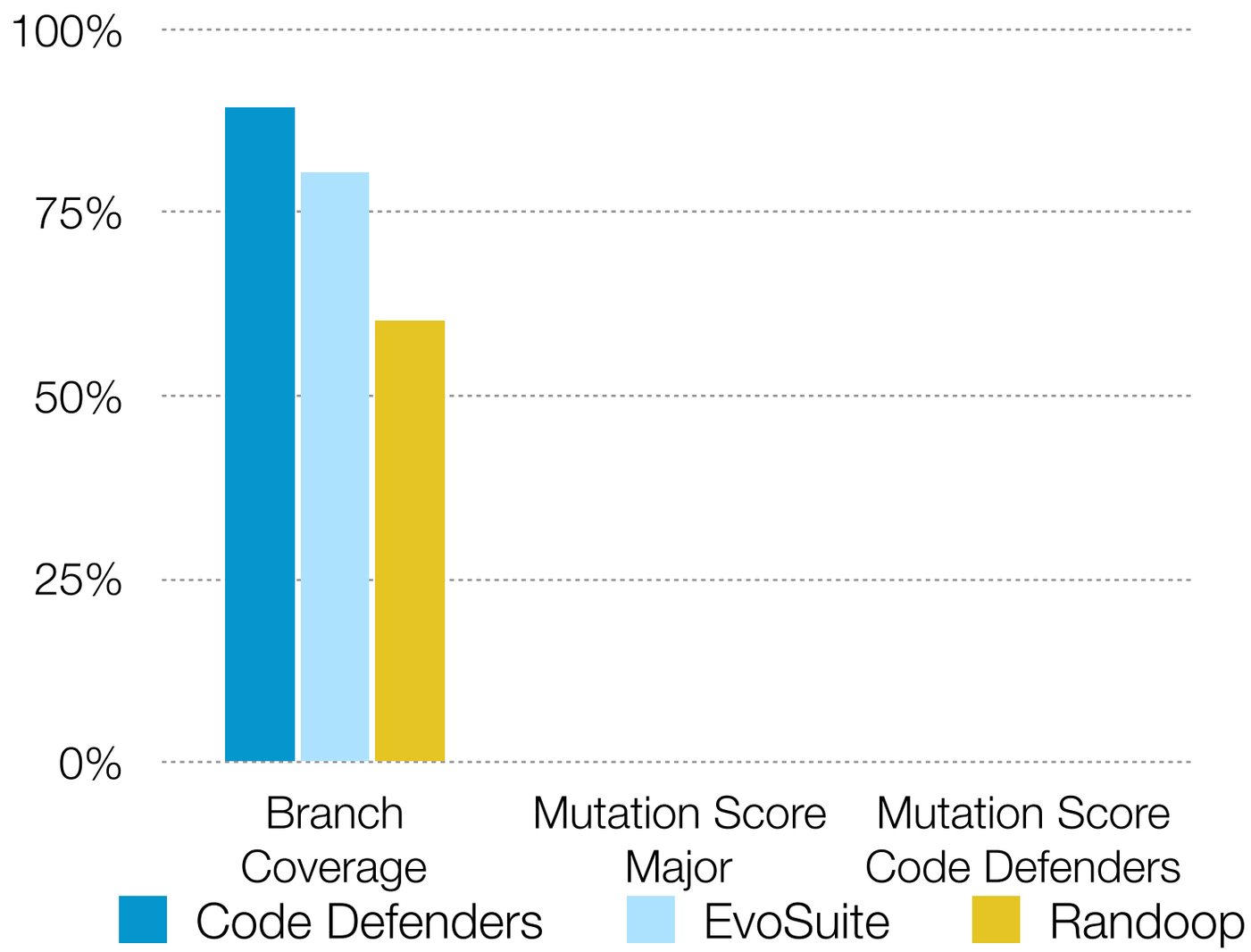


6-10

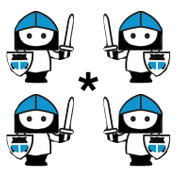
24

20

Does Crowdsourcing Work?



35

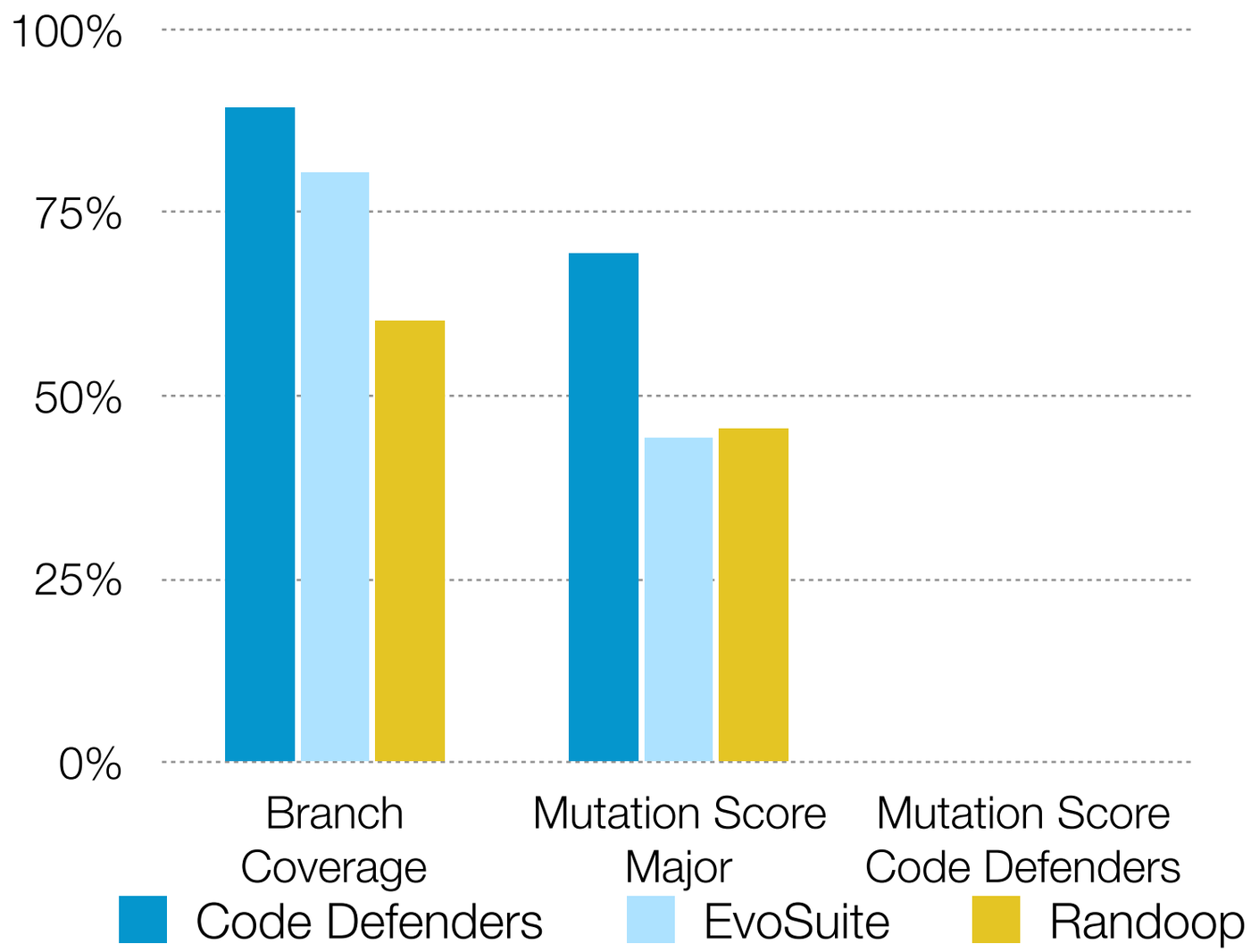


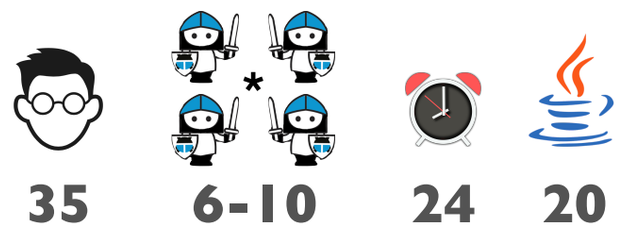
6-10

24

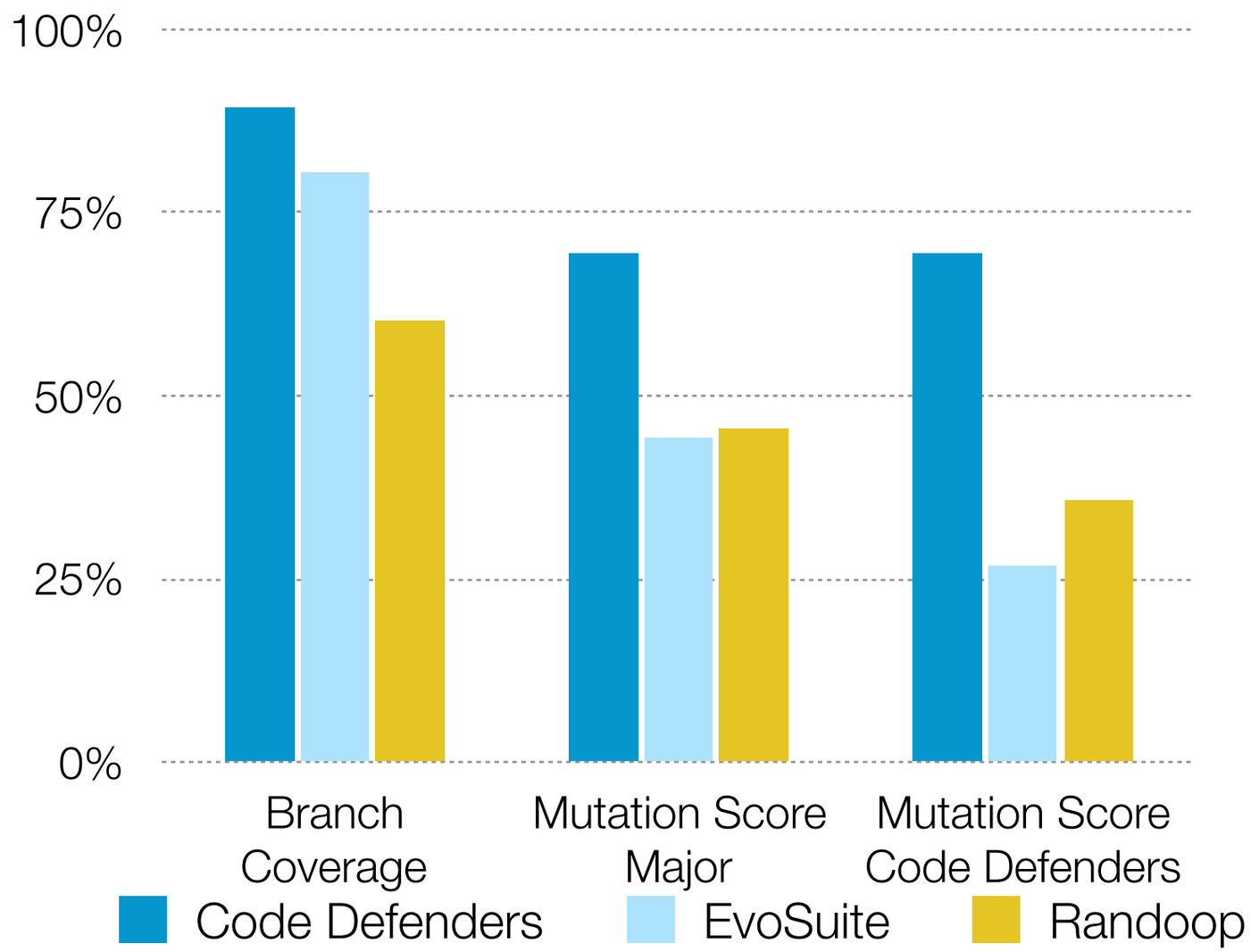
20

Does Crowdsourcing Work?



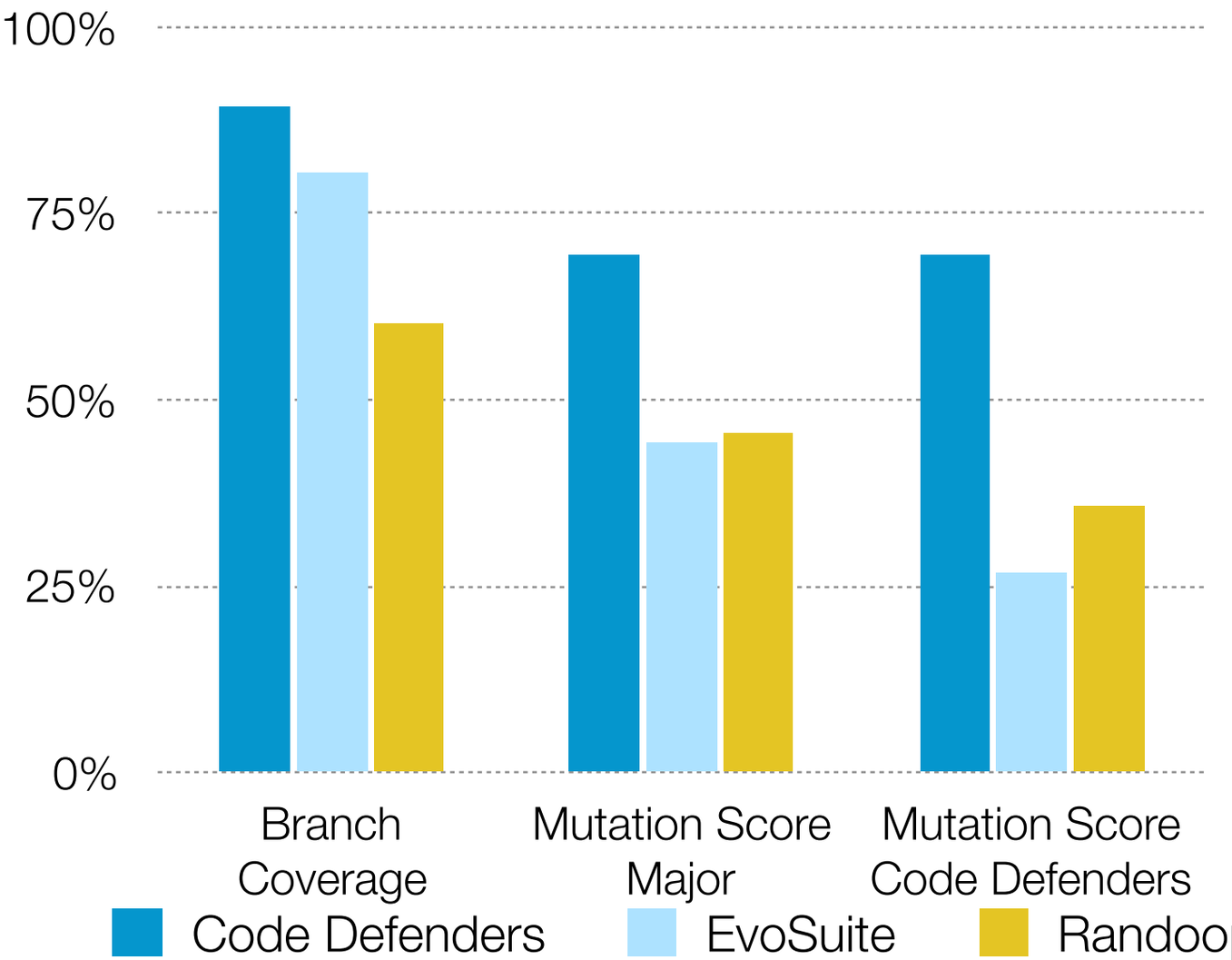


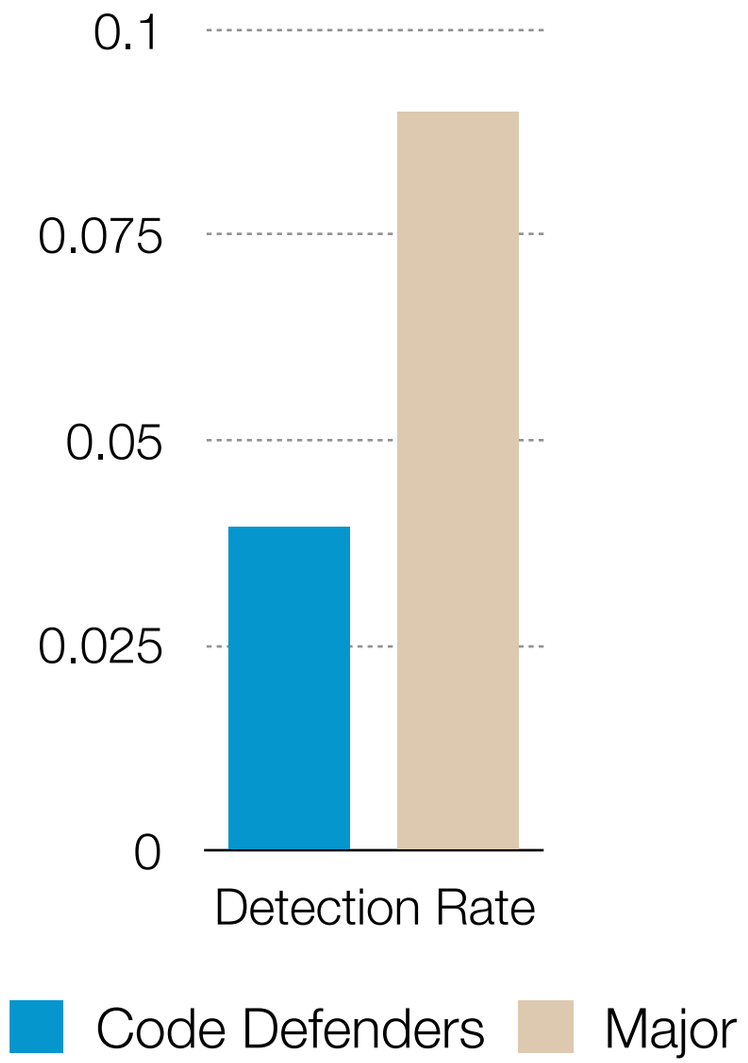
Does Crowdsourcing Work?

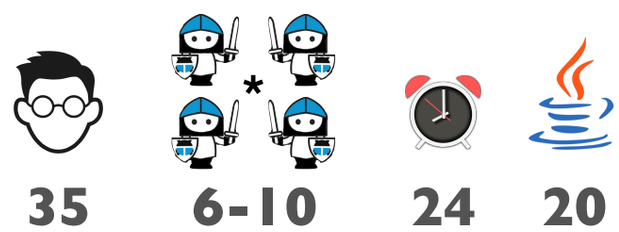


35 6-10 24 20

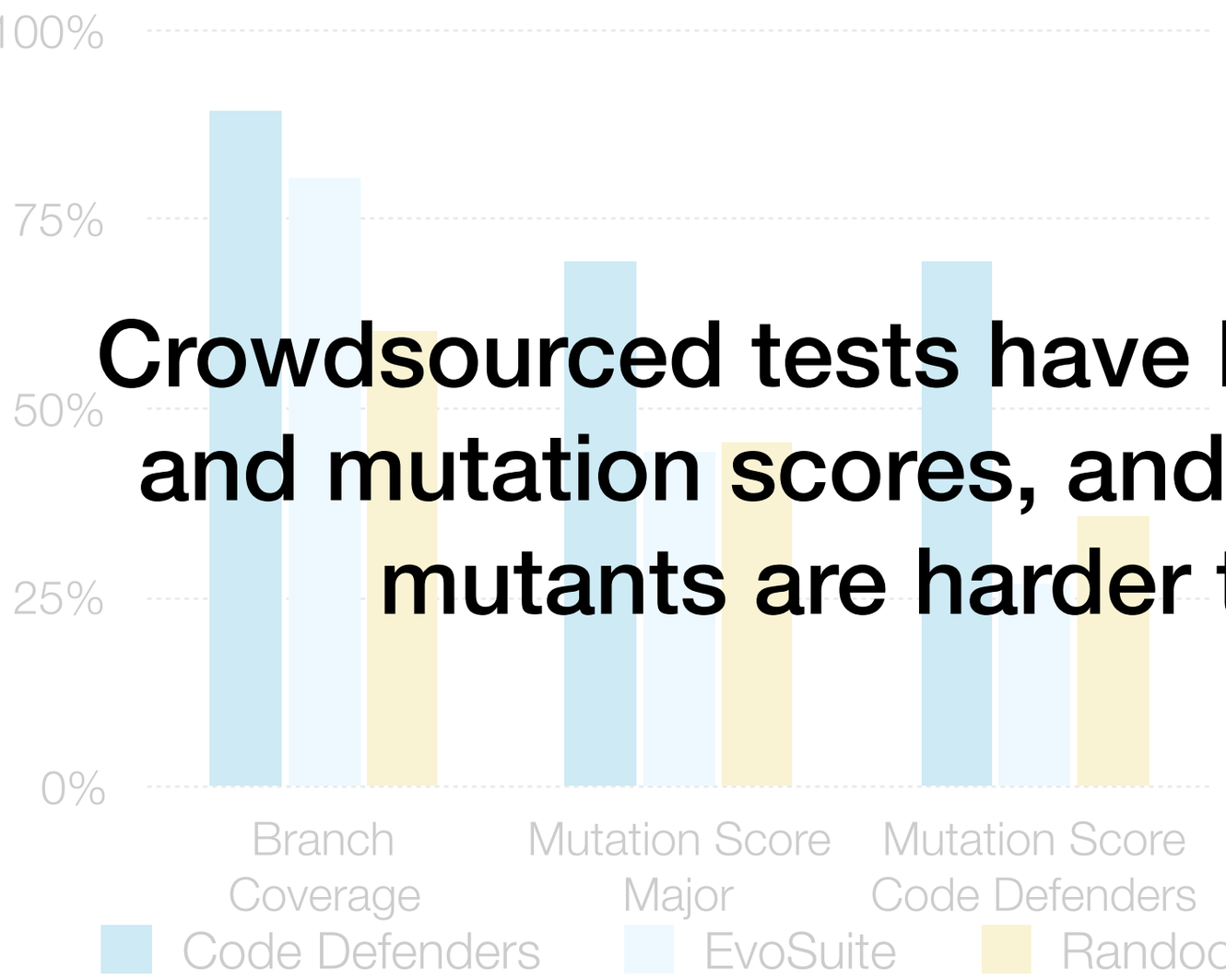
Does Crowdsourcing Work?

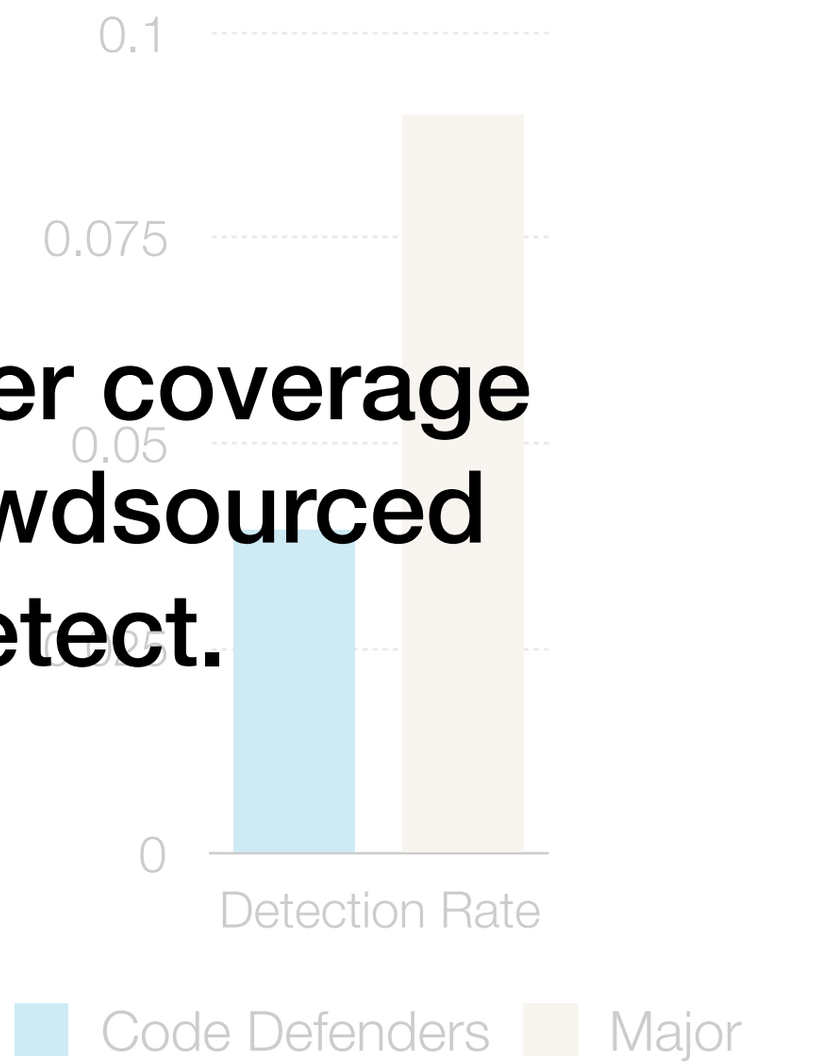




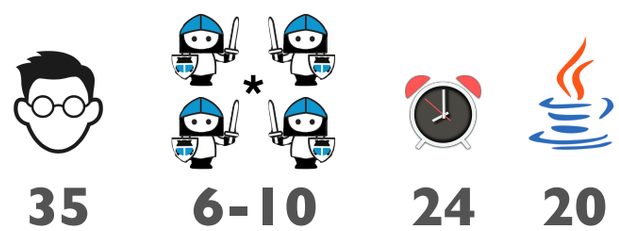


Does Crowdsourcing Work?





Crowdsourced tests have higher coverage and mutation scores, and crowdsourced mutants are harder to detect.



Challenges

Challenges

Cheating

```
public int abs(int x) {
   if (x == 8273482675) return -1;
   if (x >= 0)
      return x;
   else
      return -x;
}
```

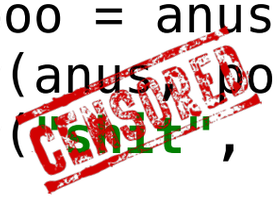
Challenges

Cheating

```
public int abs(int x) {
   if (x == 8273482675) return -1;
   if (x >= 0)
      return x;
   else
      return -x;
}
```

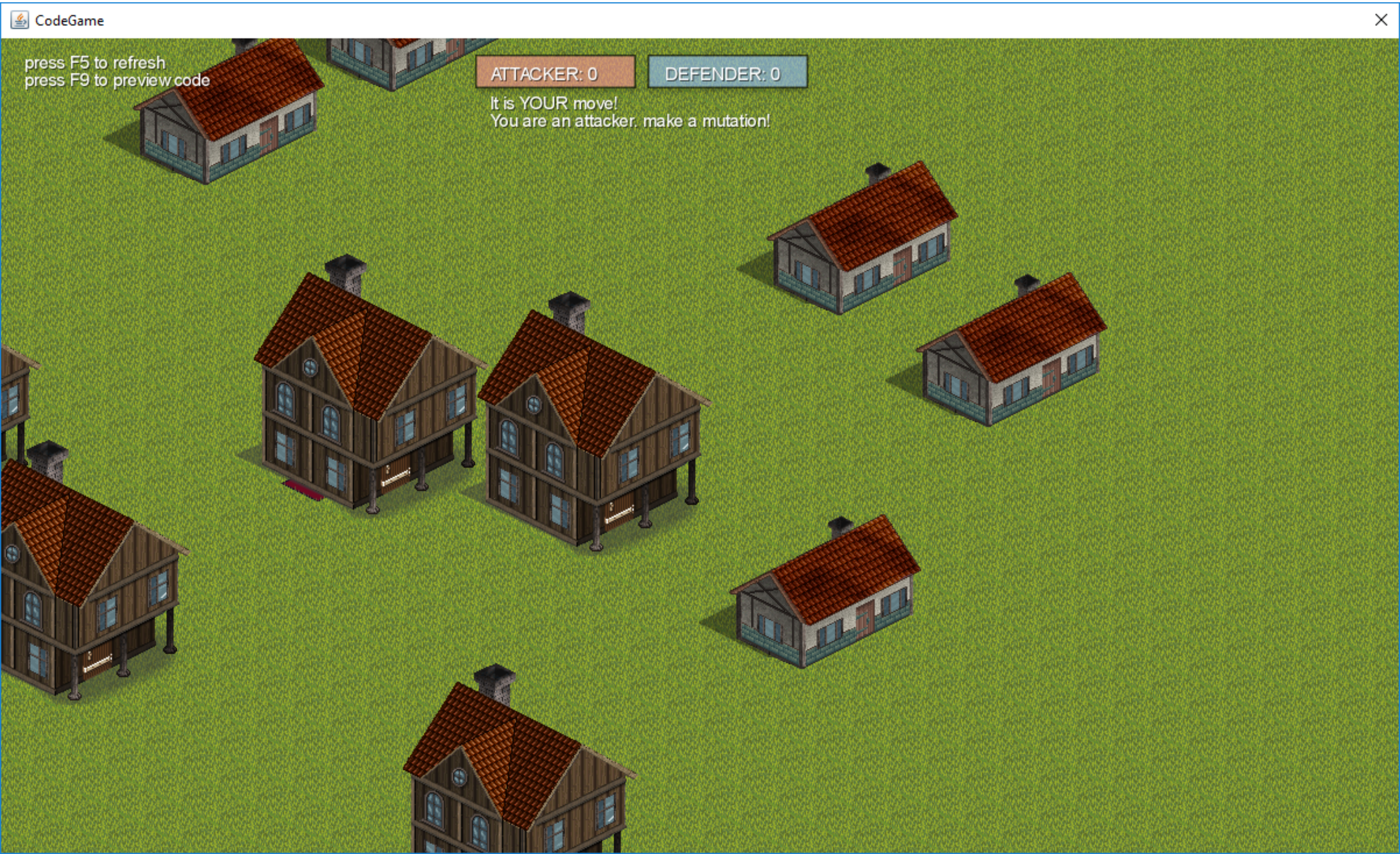
Profanity

```
@Test(timeout = 4000)
public void test() throws Throwable {
    XmlElement anus = new XmlElement();
    XmlElement poo = anus.addSubElement("shit","fuck");
    assertEquals(anus, poo.getParent());
    assertEquals("shit", poo.name);
}
```



Challenges

Narrative and other gamification elements


CodeGame
press F5 to refresh
press F9 to preview code
ATTACKER: 0
DEFENDER: 0
It is YOUR move!
You are an attacker. make a mutation!

CodeGame

press F5 to refresh
press F9 to preview code

ATTACKER: 0

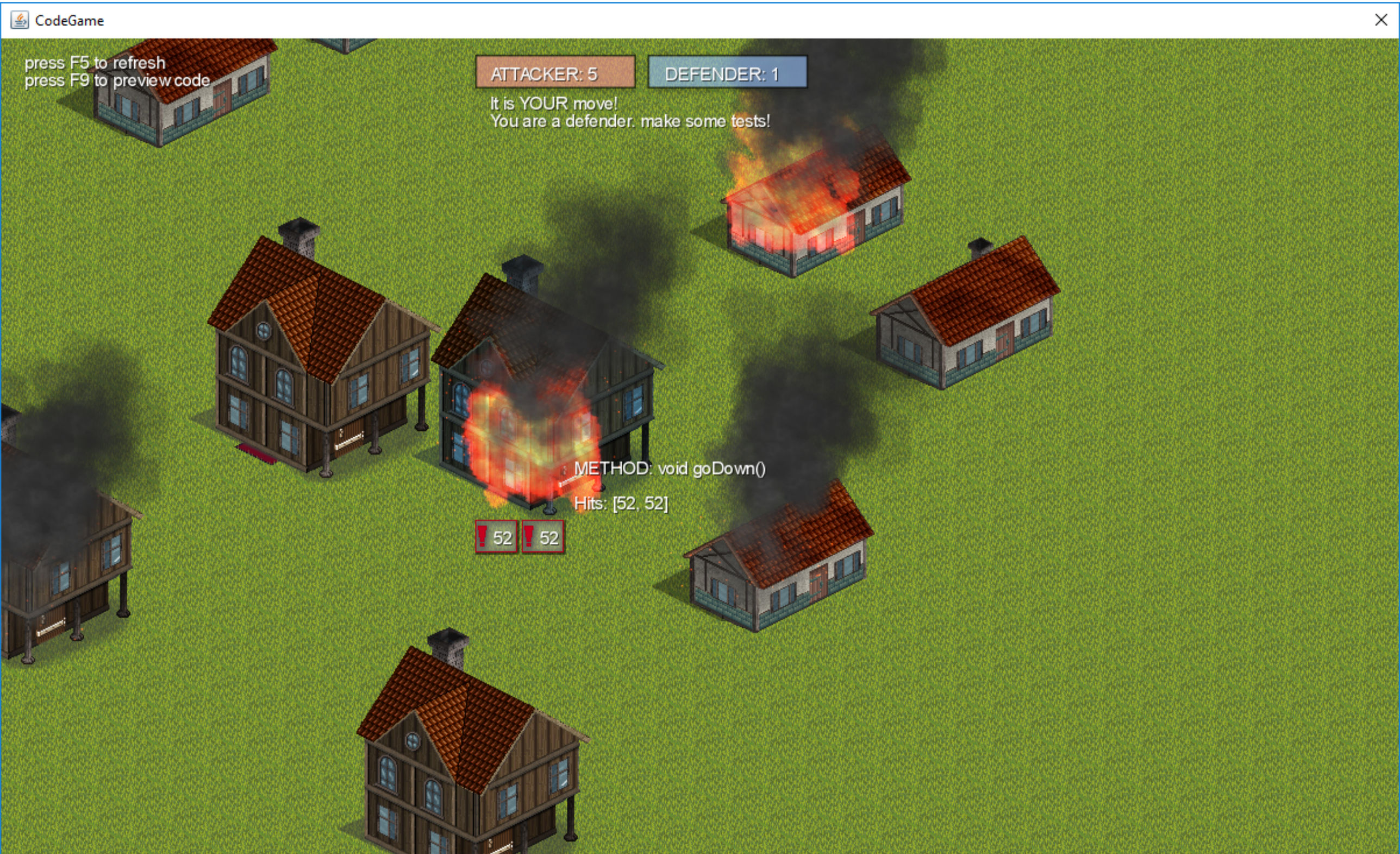
DEFENDER: 0

It is YOUR move!
You are an attacker. make a mutation!

Mutation Editor.

cancel | Create Mutation

```
1  public class Lift {
2
3      private int topFloor;
4      private int currentFloor = 0; // default
5      private int capacity = 10;    // default
6      private int numRiders = 0;    // default
7
8      public Lift(int highestFloor) {
9          topFloor = highestFloor+1;
10     }
11
12     public Lift(int highestFloor, int maxRiders) {
13         this(highestFloor);
14         capacity = maxRiders;
15     }
16
17     public int getTopFloor() {
18         return topFloor;
19     }
20
```


CodeGame
press F5 to refresh
press F9 to preview code
ATTACKER: 5
DEFENDER: 1
It is YOUR move!
You are a defender. make some tests!
METHOD: void goDown()
Hits: [52, 52]
52
52

Other Plans

Other Plans

Other Plans

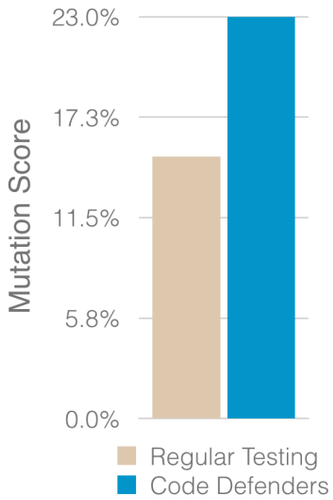
python™

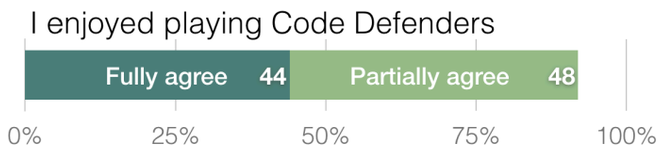
Other Plans

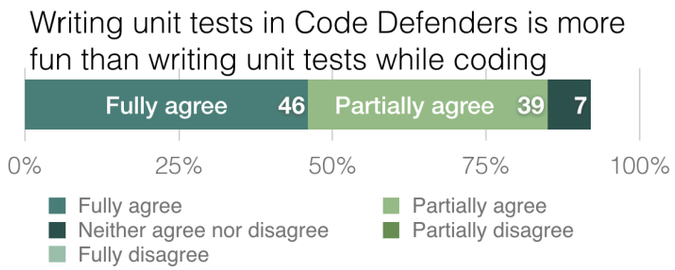
python™

Other Plans

Does Gamification Work?

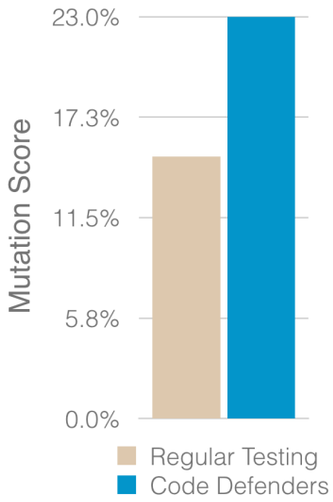


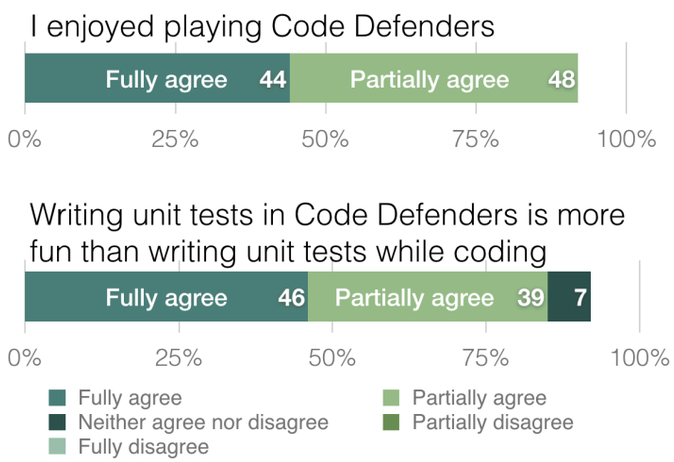




Code Defenders leads to stronger tests and is more fun than regular unit testing.

Does Gamification Work?

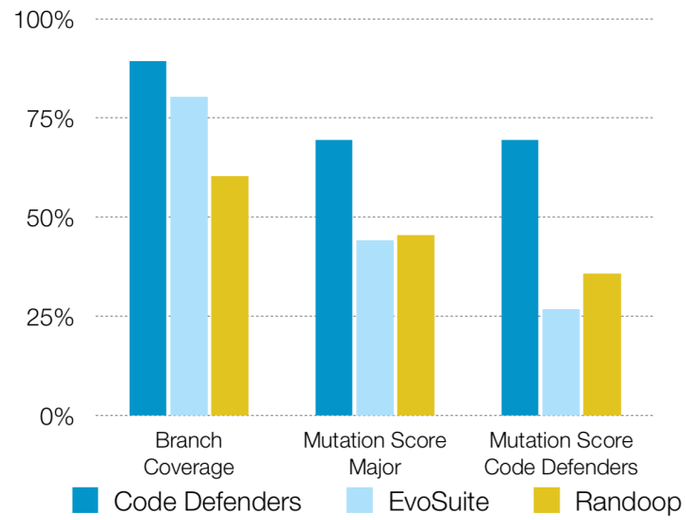


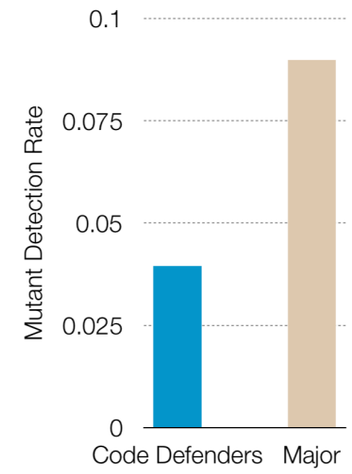


Code Defenders leads to stronger tests and is more fun than regular unit testing.

41 duels 1 2

Does Crowdsourcing Work?

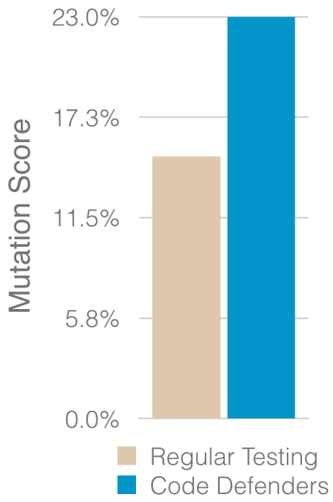


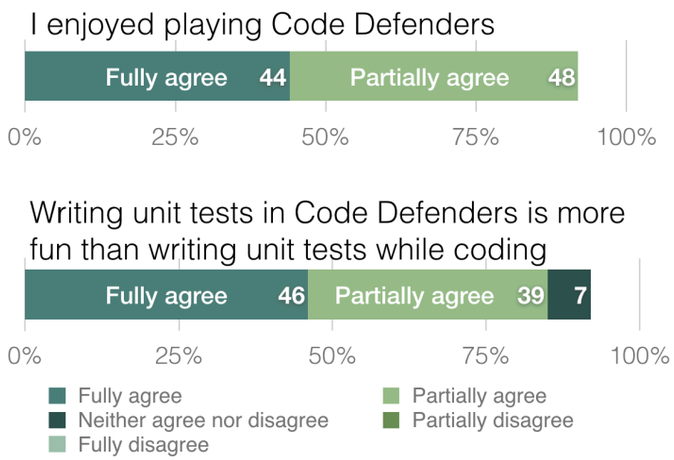


Crowdsourced tests have higher coverage/mutation scores, and crowdsourced mutants are harder to detect.

35 6-8 24 20

Does Gamification Work?

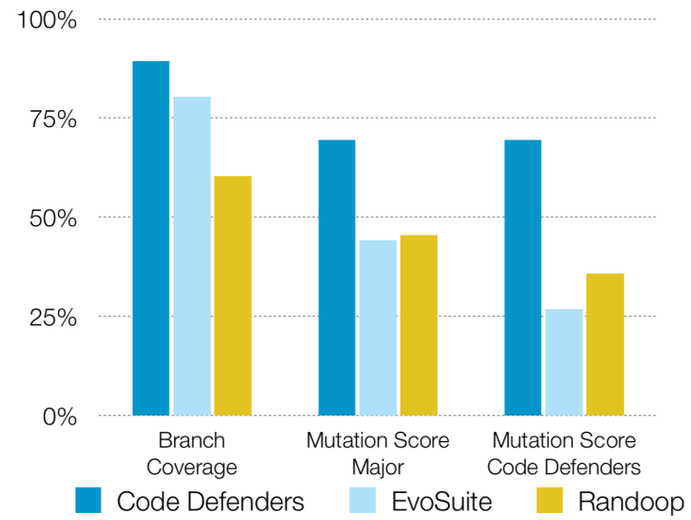


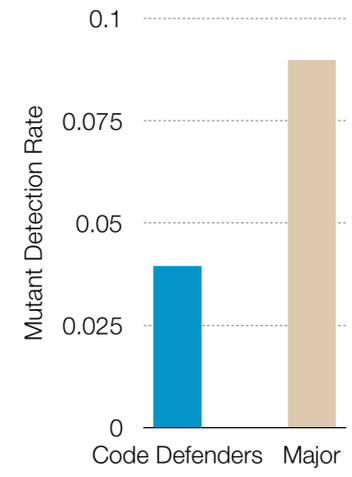


Code Defenders leads to stronger tests and is more fun than regular unit testing.

41 duels | 1 | 2

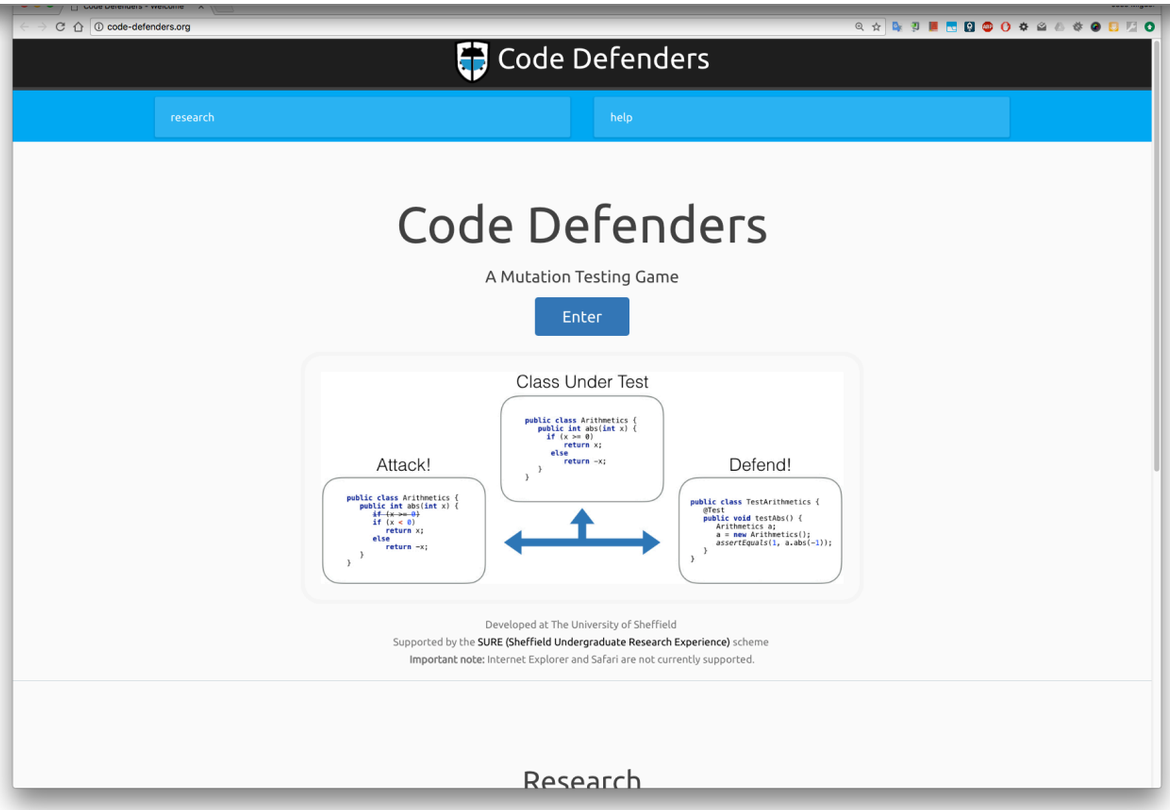
Does Crowdsourcing Work?





Crowdsourced tests have higher coverage/mutation scores, and crowdsourced mutants are harder to detect.

35 | 6-8 | 24 | 20



code-defenders.org