

STUBCODER Automated Generation and Repair of Stub Code for Mock Objects

Hengcheng ZHU¹, Lili WEI², Valerio TERRAGNI³, Yepang LIU⁴, Shing-Chi CHEUNG¹, Jiarong WU¹,
Qin SHENG⁵, Bing ZHANG⁵, and Lihong SONG⁵



Unit Testing in Practice

Unit tests need to deal with **external dependencies**.

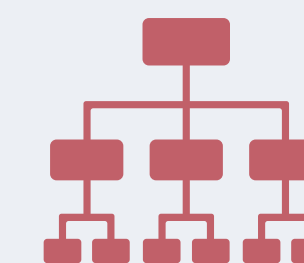
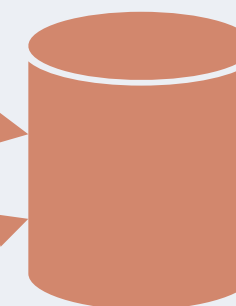
```
@Test
public void testLogin() {

    var dao = new UserDao("jdbc:...//...");
    var passwordHash = DigestUtils.sha256Hex("bar");
    var user = new User("foo", passwordHash);
    dao.insert(user);

    var authService = new AuthService(dao);
    var result = authService.login("foo", "bar");

    assertNotNull(result);
    assertTrue(result.isSuccessful());

}
```



Difficult to configure



Slow



Buggy

Mocking & Stubbing

Stubbing is to **mimic** the test dependencies in **controlled** ways.

Substitute the test dependency

```
@Test
public void testLogin() {
    var dao = mock(UserDao.class);
```

```
    var passwordHash = DigestUtils.sha256Hex("bar");
    var user = new User("foo", passwordHash);
    when(
        dao.findUser(eq("foo"))
    ).thenReturn(user);
```

Specify the behaviors

```
    var authService = new AuthService(dao);
    var result = authService.login("foo", "bar");

    assertNotNull(result);
    assertTrue(result.isSuccessful());
}
```

← Stub code

Stub Code in the Wild

Stub code requires **intensive effort** to create and maintain.

```
@Test
public void testUnbuffer() throws IOException {
    // Create mock ObjectMetadata for getFileStatus()
    Path path = new Path("/file");
    HeadObjectResponse objectMetadata = HeadObjectResponse.builder()
        .contentLength(1L)
        .lastModified(Instant.ofEpochMilli(2L))
        .eTag("mock-etag")
        .build();
    when(s3.headObject((HeadObjectRequest) any())).thenReturn(objectMetadata);

    // Create mock ResponseInputStream<GetObjectResponse> and GetObjectResponse
    GetObjectResponse objectResponse = GetObjectResponse.builder()
        .contentLength(1L)
        .lastModified(Instant.ofEpochMilli(2L))
        .eTag("mock-etag")
        .build();
    InputStream objectStream = mock(InputStream.class);
    when(objectStream.read()).thenReturn(-1);
    when(objectStream.read(any(byte[].class))).thenReturn(-1);
    when(objectStream.read(any(byte[].class), anyInt(), anyInt())).thenReturn(-1);
    ResponseInputStream<GetObjectResponse> getObjectResponseInputStream =
        new ResponseInputStream(objectResponse,
            AbortableInputStream.create(objectStream, () -> {}));
    when(s3.getObject((GetObjectRequest) any())).thenReturn(getObjectResponseInputStream);
}
```

5 stubs

```
private ContainerSimulator createContainerSim(long allocationId,
    long requestDelay) {
    TaskContainerDefinition taskContainerDef =
        mock(TaskContainerDefinition.class);
    when(taskContainerDef.getResource()).thenReturn(null);
    when(taskContainerDef.getDuration()).thenReturn(1000L);
    when(taskContainerDef.getHostname()).thenReturn("*");
    when(taskContainerDef.getPriority()).thenReturn(1);
    when(taskContainerDef.getType()).thenReturn("Map");
    when(taskContainerDef.getExecutionType()).thenReturn(null);
    when(taskContainerDef.getAllocationId()).thenReturn(allocationId);
    when(taskContainerDef.getRequestDelay()).thenReturn(requestDelay);
    return ContainerSimulator.createFromTaskContainerDefinition(
        taskContainerDef);
}
```

8 stubs

```
Parameters()
exception {
    From().thenReturn(Optional.of("base-image"));
    ExposedPorts()
    ));
    Volumes()
    .thenReturn(
        ImmutableSet.of(AbsoluteUnixPath.get("/volume1"), AbsoluteUnixPath.get("/volume2")));
    when(mockCommonContainerConfigCliOptions.getEnvironment())
        .thenReturn(ImmutableMap.of("key1", "value1"));
    when(mockCommonContainerConfigCliOptions.getLabels())
        .thenReturn(ImmutableMap.of("label", "mylabel"));
    when(mockCommonContainerConfigCliOptions.getUser()).thenReturn(Optional.of("customUser"));
    when(mockCommonContainerConfigCliOptions.getFormat()).thenReturn(Optional.of(ImageFormat.OCI));
    when(mockCommonContainerConfigCliOptions.getProgramArguments())
        .thenReturn(ImmutableList.of("arg1"));
    when(mockCommonContainerConfigCliOptions.getEntrypoint())
        .thenReturn(ImmutableList.of("custom", "entrypoint"));
    when(mockCommonContainerConfigCliOptions.getCreationTime())
        .thenReturn(Optional.of(Instant.ofEpochSecond(5)));
}
```

10 stubs

Code snippets are adopted from
- <https://github.com/apache/hadoop>
- <https://github.com/GoogleContainerTools/jib>

STUBCODER

Generates stub code to **pass the test**. For **regression testing** scenario.


```
@Test
public void testLogin() {

    var dao = mock(UserDao.class);
```

w/o stub code

w/ obsolete stub code

```
var authService = new AuthService(dao);
var result = authService.login("foo", "bar");
```

```
 assertNotNull(result);
assertTrue(result.isSuccessful());
}
```


Generate

```
@Test
public void testLogin() {
```

```
var dao = mock(UserDao.class);
```

```
var passwordHash = DigestUtils.sha256Hex("bar");
var user = new User("foo", passwordHash);
when(
    dao.findUser(eq("foo"))
).thenReturn(user);
```

```
var authService = new AuthService(dao);
var result = authService.login("foo", "bar");
```

```
 assertNotNull(result);
assertTrue(result.isSuccessful());
}
```

STUBCODER - Evolutionary Algorithm

Fitness function measures **how far the test case is from passing.**

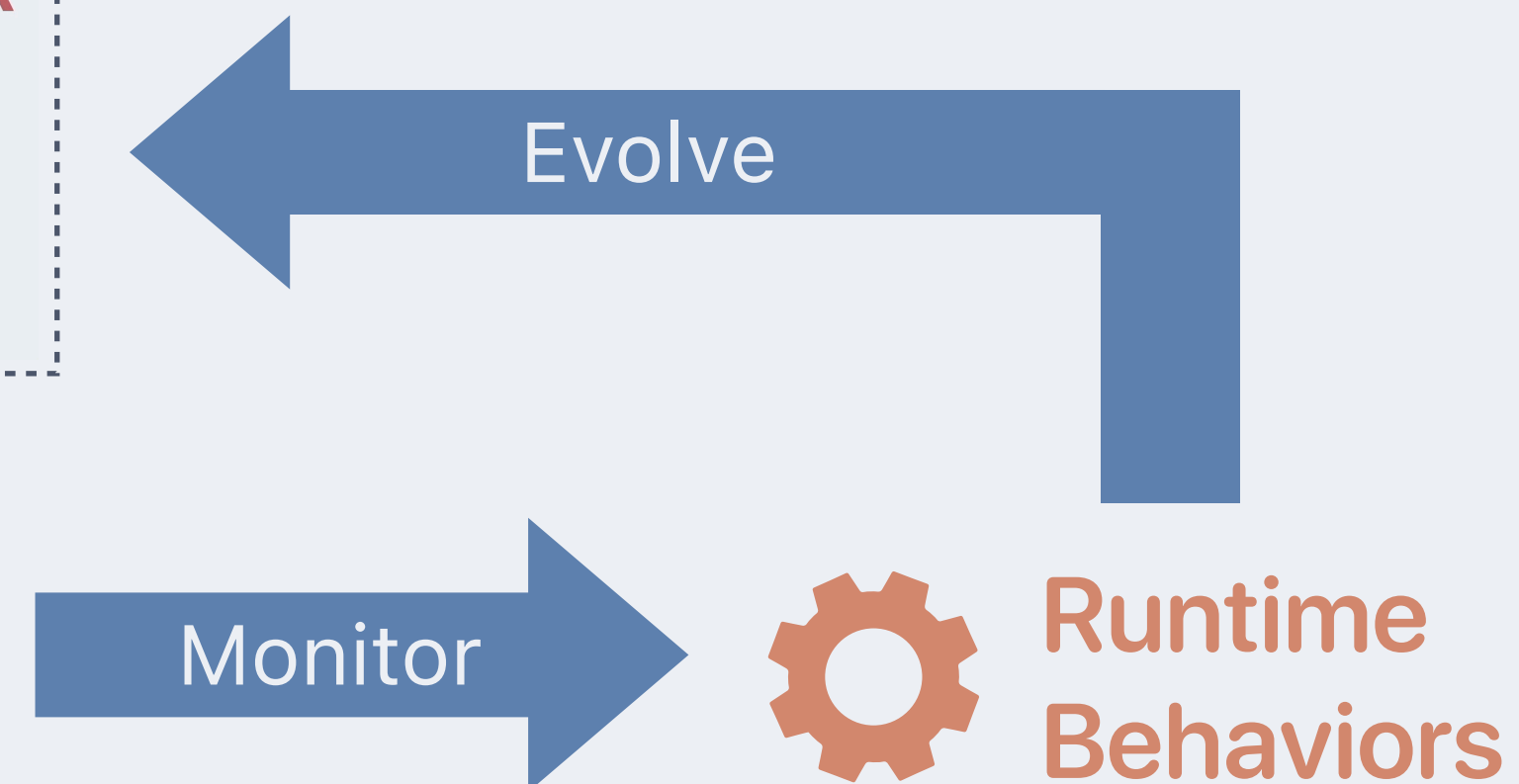
```
@Test  
public void testLogin() {
```

```
    var dao = mock(UserDao.class);
```

```
vuN}s5)4b@n:\e]<50QWEycE>&#Mwks>Y|]F/xxW}<&D+YLR  
vwNNj6-:=&\to_.'v|=\_?3+rbn5<Ko7iNt6:|d0]  
w2qmm  
z0rYb>tfe7sR;J6d1:B<wx;%  
)=_0!C;5>Tf.x0jib3p
```

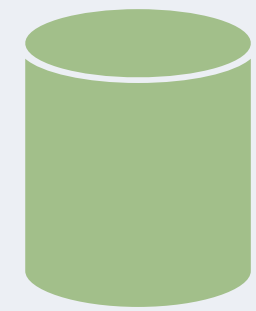
```
var authService = new AuthService(dao);  
var result = authService.login("foo", "bar");  
  
assertNotNull(result);  
assertTrue(result.isSuccessful());
```

```
}
```

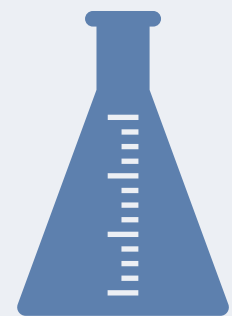


Evaluation

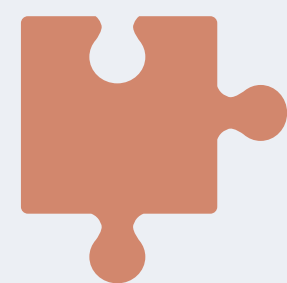
Evaluation Subjects



13 Open-source
Projects



59 Test cases



167 Mock objects

Evaluation Results

76%

Success Rate

StubCoder produces stub code leading to a passing test

99%

Similarity in Execution Path

Similarity between execution path with developer written test and StubCoder generated test (median)

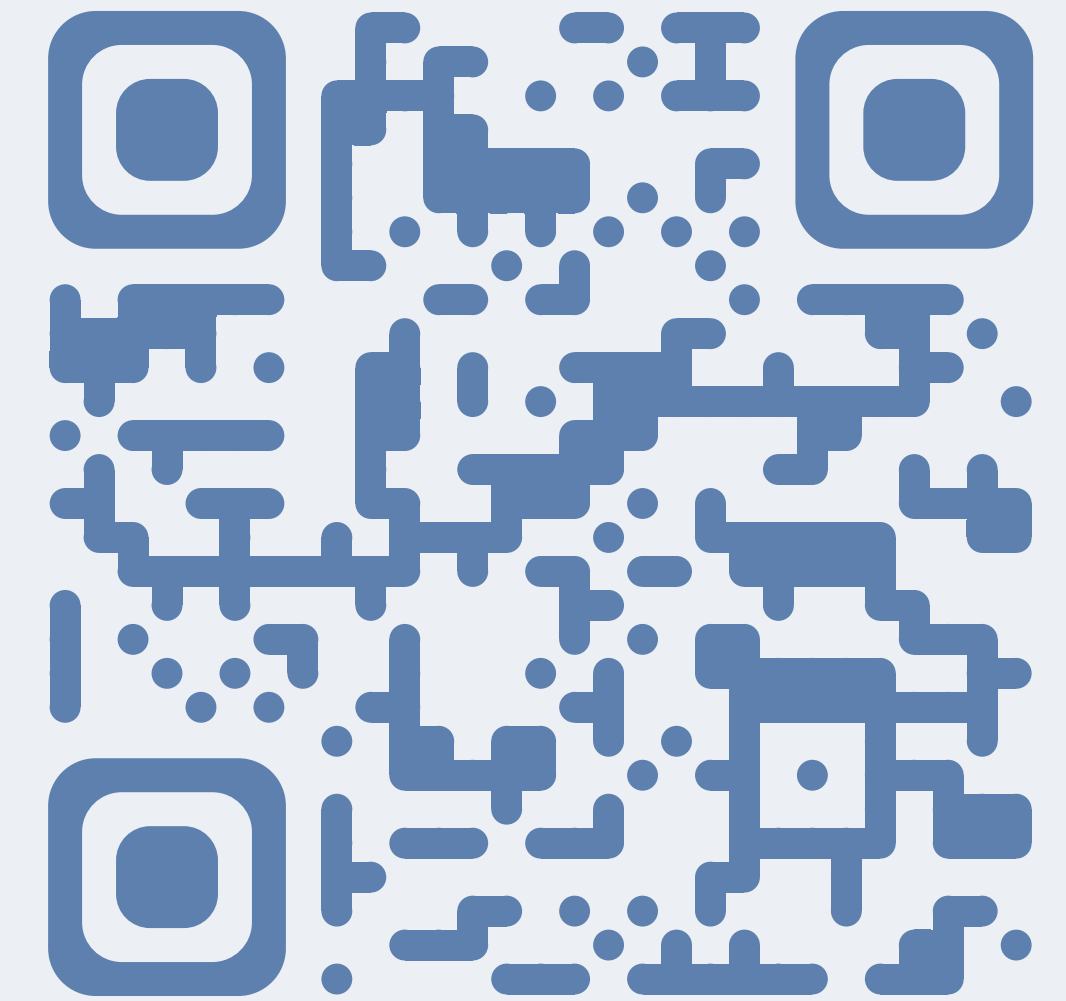
100%

Similarity in Killed Mutants

Similarity between the set of killed mutants with developer written test and StubCoder generated test (median)

Thanks

Hengcheng ZHU hzhuaq@connect.ust.hk



aka.henryhc.net/stubcoder

