CSE 250
Lecture 3

How to Scala (… when things go wrong™)

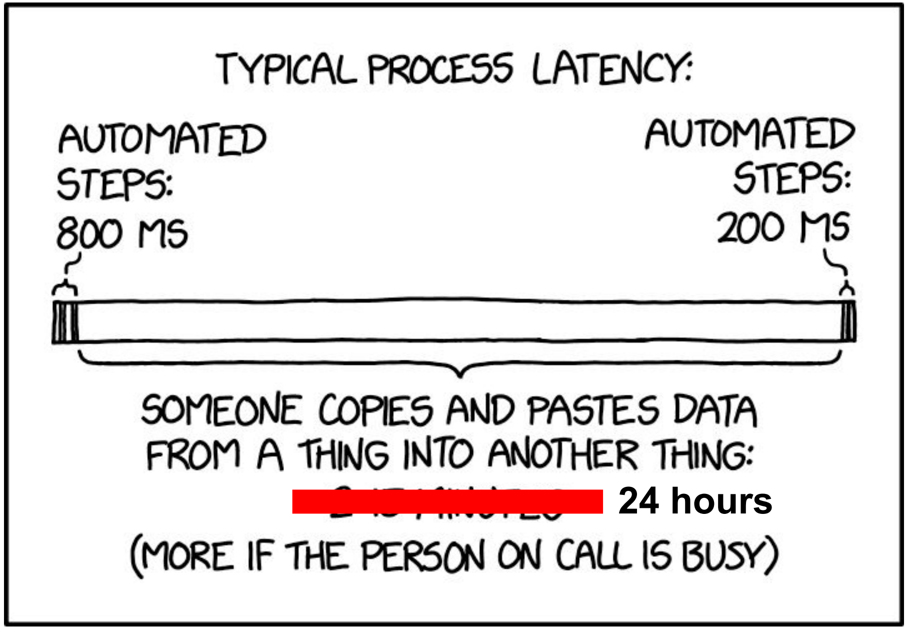
Announcements

- AI Quiz on Autolab
  - Due Wednesday Night (~ 10 minutes)
- PA 0 (aka “set up Git and give us your Github username”) on Autolab
  - Due 1 week from today (< 30 minutes)
- PA 1 (aka “Parse CSV files in Scala”) on Autolab
  - Due 2 weeks from today
  - Submissions open a week from today
  - Start Early! (PA 1 template includes starter test cases)

https://xkcd.com/2565/

GitHub + Autolab

- Help us open submissions early:
  - We’ll start them once ~90% of Sections A, B submit PA 0
- Allow 24 hours after submitting PA0 before you can submit PA1


TYPICAL PROCESS LATENCY:
AUTOMATED STEPS: 800 MS
AUTOMATED STEPS: 200 MS
SOMEONE COPIES AND PASTES DATA FROM A THING INTO ANOTHER THING:
[redacted] 24 hours
(MORE IF THE PERSON ON CALL IS BUSY)

Testing

The REPL

- From IntelliJ: `Ctrl+Shift+D`
  - Highlight a line and press `Ctrl+Shift+X` to `execute`
  - Copy+paste line and press `Ctrl+Enter` to `execute`

- From the command line: `scala`
  - Paste commands in to run them
  - Type `:help` to get a list of extra commands

- From SBT: `console`

Unit Testing

- Remember breaking the big problem into smaller ones?
  - Test each small solution before combining into a whole

- Useful for debugging
  - Sanity check each step in the process to make sure it works
  - Separate the UI from the tests

- Useful way to encode your assumptions
  - Helpful reminder if your assumptions change later

Basic Debugging

Demo

ScalaTest

```
class HelloWorldTest extends AnyFlatSpec {
  "HelloWorld.doThings()" should "return 5" in {
    assert(HelloWorld.doThings() == 5)
  }
  it should "not return 10" in {
    assert(HelloWorld.doThings() != 10)
  }
  "HelloWorld.x" should "have type Float" in {
    assert(HelloWorld.x.isInstanceOf[Float])
  }
  "Register(0).addToValue" should "return the input value"
  in {
    val reg = Register(0)
    for (i <- 1 to 10000) { assert(reg.addToValue(i) == i) }
  }
}
```

Describe the test in English

‘in’ block defines the tests

Confirm assumptions with assert

Call assert as many times as necessary

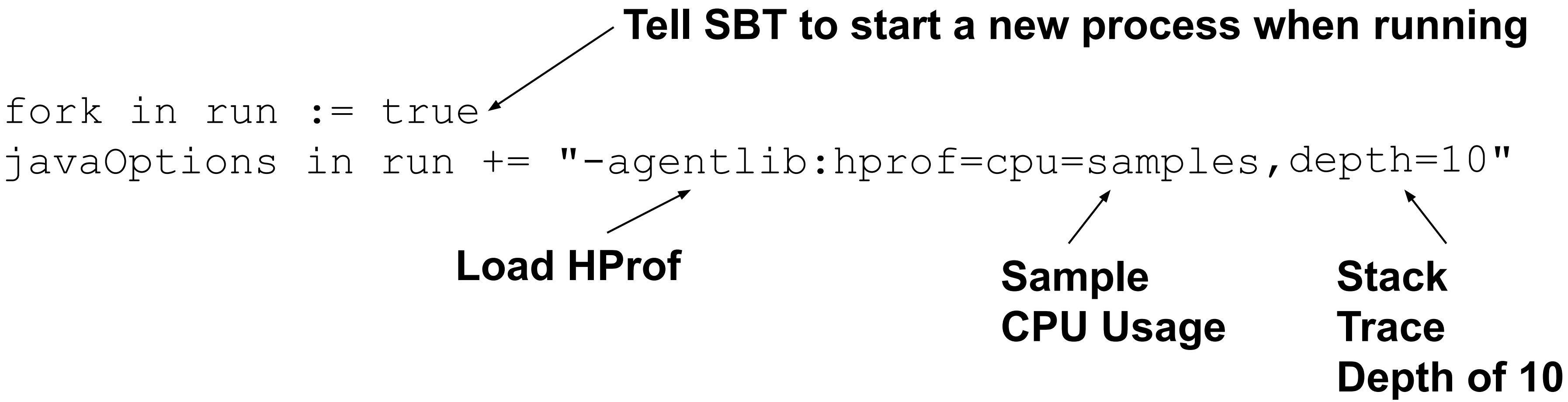
Debugging with ScalaTest

Demo

Profiling

- IntelliJ → Profilers
  - https://www.jetbrains.com/help/idea/cpu-profiler.html

- SBT → HProf
  - https://docs.oracle.com/javase/8/docs/technotes/samples/hprof.html



Tell SBT to start a new process when running

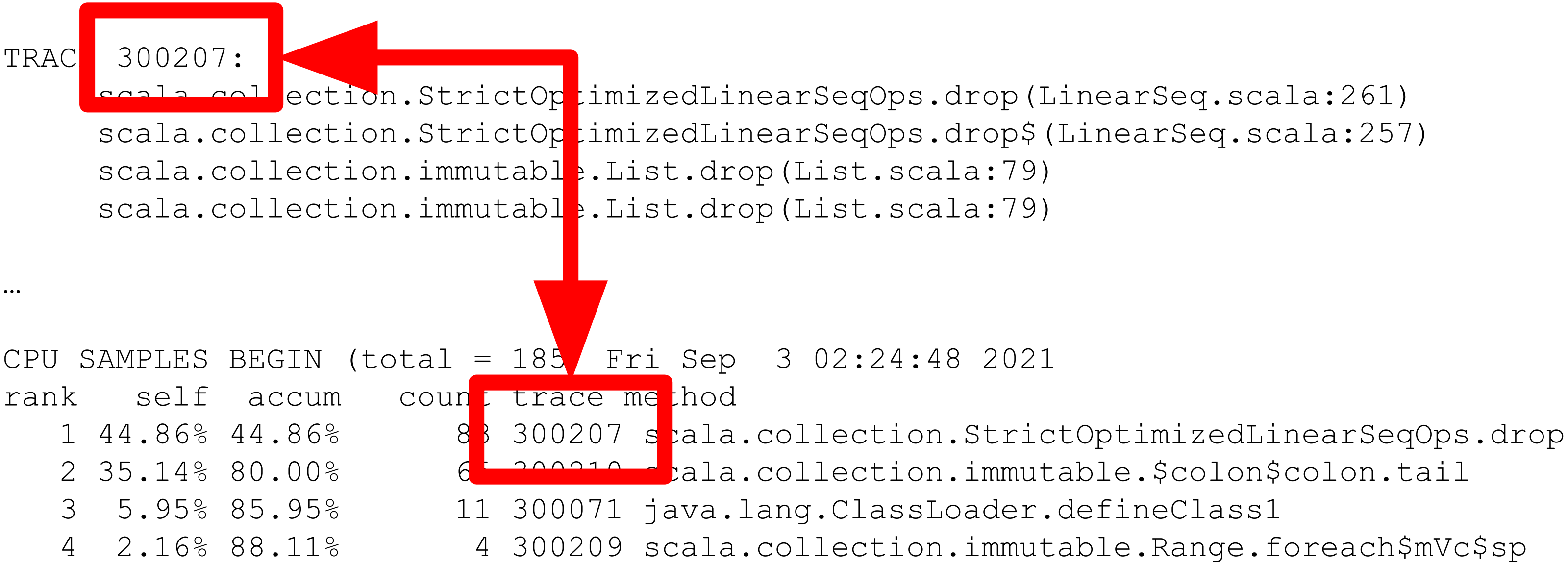
```
fork in run := true
javaOptions in run += "-agentlib:hprof=cpu=samples,depth=10"
```

Load HProf

Sample CPU Usage

Stack Trace Depth of 10

HProf Traces



```
JAVA PROFILE 1.0.1, created Fri Sep  3 02:24:46 2021

Copyright (c) 2003, 2005, Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without

…

TRACE 300207:
     scala.collection.StrictOptimizedLinearSeqOps.drop(LinearSeq.scala:261)
     scala.collection.StrictOptimizedLinearSeqOps.drop$(LinearSeq.scala:257)
     scala.collection.immutable.List.drop(List.scala:79)
     scala.collection.immutable.List.drop(List.scala:79)

…

CPU SAMPLES BEGIN (total = 185) Fri Sep  3 02:24:48 2021
rank   self  accum   count trace method
   1 44.86% 44.86%      83 300207 scala.collection.StrictOptimizedLinearSeqOps.drop
   2 35.14% 80.00%      65 300210 scala.collection.immutable.$colon$colon.tail
   3  5.95% 85.95%      11 300071 java.lang.ClassLoader.defineClass1
   4  2.16% 88.11%       4 300209 scala.collection.immutable.Range.foreach$mVc$sp
```

Profiling with HProf

Demo

Submission

- **Github Classroom**
  - You will get an invitation link for each individual project.
  - Clicking the invitation will create a GitHub Git repository for the project with template code.
- Edit the GitHub classroom repository with your submission
  - Create a new submission in Autolab when ready.
- Requirements
  - Make sure you are using Scala 2.13.x
  - Don’t add any outside packages