Laboratory: Scala (3)

Summary: We conclude our exploration of the Scala programming language, focusing on its XML, pattern matching, and concurrency features.

Prerequisites: The first Scala lab. The second Scala lab. Tate, Sections 5.4 and 5.5.

Contents:

- Preparation
- Exercises
  - Exercise 1: Fun with XML
  - Exercise 2: Tables of Contents
  - Exercise 3: Printing Links
  - Exercise 4: Tate’s Page Sizing Algorithm
  - Exercise 5: Web Crawling
- For Those with Extra Time

Preparation

a. Create a directory for the lab.

b. Open a browser window on Tate’s examples, in case you want to try any of them.

Exercises

Exercise 1: Fun with XML

a. Try Tate’s movies/shorts example, from the section entitled “XML with Matching”.

b. Write an expression to print just the shorts in the collection of movies.

c. To the full movies, add subsections for chapters. E.g.,

```xml
<movie>The Incredibles
  <chapter>Opening</chapter>
  <chapter>Heroes in Love</chapter>
  <chapter>Oh No! It’s the bad guy.</chapter>
</movie>
```

d. What do you expect the effect to be if you run Tate’s `foreach` matching expression using this expanded collection?
e. Check your answer experimentally?

f. To the list of movies, add a series type. E.g.,

    <series>Glee, Season 1</series>


g. What do you expect the effect to be if you run Tate’s foreach matching expression using this expanded collection?

h. Check your answer experimentally.

**Exercise 2: Tables of Contents**

Here is a sketch of a simple HTML page.

```html
<html>
<head>
<title>A Sample Page</title>
</head>
<body>
<h1><a name="top">A Sample Page</a></h1>
<h2><a name="sect1">Section 1</a></h2>
<h2><a name="sect2">Section 2</a></h2>
</body>
</html>
```

Write instructions to generate a simple “table of contents” for this page. That is, for each anchor target, you should generate a link. The result of this might be:

```
<ul>
<li><a href="#top">A Sample Page</a></li>
<li><a href="#sect1">Section 1</a></li>
<li><a href="#sect2">Section 2</a></li>
</ul>
```

It would, of course, be better to generate a hierarchical table of contents. However, that might be significantly more work and places more assumptions on the structure of the original.

**Exercise 3: Printing Links**

Write a script that takes an HTML page and prints out all of the links in that page.

**Exercise 4: Tate’s Page Sizing Algorithm**

a. Try Tate’s algorithm for figuring out the sizes of a group of pages.

b. See if you can figure out a way to get a sum of sizes out of the algorithm.
Exercise 5: Web Crawling

Write a script that takes an HTML page and prints out the size of all the linked files in that page. Make sure that you fetch the linked files concurrently.

For Those with Extra Time

If you find yourself with extra time, continue working on Assignment 5.