Prof. P. Thiemann   Sommer Term 2009
M. Mehlmann

Software Engineering

Exercise Sheet 2

Given the following Featherweight Java program:

```
class Author extends Object {
  String firstName;
  String lastName;

  Author(String firstName, String lastName) {
    super();
    this.firstName = firstName;
    this.lastName = lastName;
  }
}

class Book extends Object {
  Author author;

  Book(Author author) {
    super();
    this.author = author;
  }

  String getAuthorLastName() {
    return this.author.lastName;
  }
}

class BestsellerBook extends Book {
  int howManySold;
  BestsellerBook(Author author, int howManySold) {
    super(author);
    this.howManySold = howManySold;
  }
}
```

Exercise 1: Evaluating Featherweight Java (4 Points)

We liberally extend Featherweight Java with support for strings: The class
`String` is the type for string literals of the form `"This is some string"`. Now
evaluate the following expression:

```
new Book(new Author("Benjamin", "Pierce")).getAuthorLastName()
```

List all intermediate results and explain for every reduction step which reduction rule you have used.

Exercise 2: Type checking Featherweight Java (6 Points)

To extend Featherweight Java with support for strings and integers, we need two new typing rules.

$$\frac{\text{STRING} \atop s \text{ is a string literal}}{A \vdash s : \texttt{String}} \qquad \frac{\text{INT} \atop i \text{ is an integer literal}}{A \vdash i : \texttt{Int}}$$

Now give a typing derivation for the following expression:

```
new BestsellerBook(new Author("Benjamin", "Pierce"), 1024).
                   getAuthorLastName()
```