Meeting 22 - Encapsulating Effects

Bor-Yuh Evan Chang

Thursday, November 7, 2024

Meeting 22 - Encapsulating Effects

What questions does your neighbor have?

Links

In-Class Slides

In-Class Jupyter

Book Chapter

Announcements

- Remainder of the Semester
  - HW 5 and Lab 5 before Thanksgiving break
  - Unit 6 (probably one combined assignment) after Thanksgiving break
  - Exam 5-6 in the last week of classes before the Final
  - Possible extra credit project opportunity
- Exam 3-4 grading in progress
  - See how the Review Exercises/Practice Exam translates to the exam.
  - Some questions were even exactly the same (1e on giving the small-step rules for `const x = e1; e2`)
- Come see us to make a study plan
  - e.g., via the redo policy
  - see the Final Exam as an opportunity to show growth from mid-semester exams.
  - 25 minutes per unit x 6 units = 150 minutes

Today

- Encapsulating Effects
- Triage Your Questions
  - Exam 3-4

Questions?

- Review:
  - What is soundness (or unsoundness) of static typing?

Questions?

Error Effects

Exceptions

```
defined function toDoubleException
res0_1: Double = 1.0
res0_2: Double = 4.2
```

Option

```
defined function toDoubleOption
defined function toDoubleNoNaNOption
```

Option as a Collection

```
defined function toDoubleNoNaNOption
defined function addToDoubleOption
```

Map, Filter, FlatMap

Exercise 1 Implement `map` for `Option[A]`s:

```
defined function map
```

Exercise 2 Implement `filter` for `Option[A]`s:

```
defined function filter
```

Exercise 3 Implement `flatMap` for `Option[A]`s:

```
defined function flatMap
```

Comprehensions

```
defined function addToDoubleOption
```

Either and Try

```
defined function toDoubleEither
defined function addToDoubleEither
import scala.util.Try
defined function toDoubleTry
defined function addToDoubleTry
```

Mutation Effects

```
defined function freshVarImperative
defined function freshVar
```

Encapsulating Mutation Effects

```
defined type DoWith
res25: Int => (Int, String) = ammonite.$sess.cmd22$Helper$$Lambda$2244/0x000000
```

```
defined function map
defined function flatMap
counter: Int = 0
```

DoWith

```
defined class DoWith
defined object DoWith
import DoWith._
defined function freshVar
counter: Int = 0
```