

Types and Programming Languages week1

Michelle Bergin

October 14, 2018

1 Do the exercises at the end of the Ch03a lecture notes

1. Draw an abstract syntax tree for the following UAE terms (see item 2 above for the abstract syntax of UAE)

- (a) $t1 == \text{if false then } 0 \text{ else } (\text{succ } 0)$

```
      if
     /  \
  false  true
   |      |
  then   else
   |      |
   0      (succ 0)
```

- (b) $t2 == \text{if } (\text{iszero } (\text{pred } (\text{succ } 0))) \text{ then } (\text{pred } (\text{pred } 0)) \text{ else } (\text{succ } (\text{succ } 0))$

```
      if
     /   \
  true   false
```

2. Exercise in item 3 above

- (a) Give an abstract syntax for the λ calculus. Use App for the apply operator and use "Abs x" for the abstraction operator with variable x. See Selinger Ch 2.1 for the concrete syntax.
- (b) Give the abstract syntax tree for the λ expression apply3x where $\text{apply3x} == \lambda f. \lambda x. f(f(fx))$

\

```

      |
    fx
      |
      @
      |
      f
      |
      @
      |
      f
      |
      @
      |
      f
      |
    Abs
      |
      x

```

- (c) Give the abstract syntax tree for the lambda expression apply2x where $\text{apply2x} == \lambda g.\lambda x.g(gx)$
- (d) Give the abstract syntax tree for the lambda expression $(\text{apply3x } \text{apply2x})$
- 3. Draw an abstract syntax tree for the UAE terms t_1 and t_2 in item 12 above using the Haskell abstract syntax.
- 4. Exercise in item 14 above.
Rewrite the Pierce Lemma 3.3.3 proof as a structural induction proof following the style of the Pierce 3.3.4 discussion.
- 5. Pierce Exercise 3.2.4
- 6. Pierce Exercise 3.2.5