Week 7 Recitation

1. (*LetRec*) Write the abstract syntax for the following lettuce program:

```
let rec f = function(x) if x == 0 then 0 else f(x - 1) in
  f(2)
```

2. (*Closures*) What does the following program evaluate to under the current rules?

```
let y = 3 in
  let f = function(x) x + y in
    let y = 4 in
      f(2)
```

3. (*Closures*) You might be wondering why we're going to all this trouble with Closures, why we don't just leave out the environment capturing all together. What does the example from 2 evaluate to under this updated (in blue) rule which ignores the closed environment?

$$\frac{\sigma \vdash e_f \Downarrow \mathtt{Closure}(p, e_b, \pi) \quad \sigma \vdash e_a \Downarrow v_a \quad \sigma\,[p \mapsto e_a] \vdash e_b \Downarrow v \quad v_a \not= \mathtt{Error}}{\sigma \vdash \mathtt{FunCall}(e_f, e_a) \Downarrow v}\ (\Downarrow\text{-FunCall-ok})$$

4. (*Functions and derivations*) Write out the derivation for the following program:

```
(function(x) x)(3)
```