Due date: Oct 15th in class.

1. (5 points). For the following program,

\[
[x := 1]; (\text{while } [y > 0] \text{ do } [x := x - 1]); [x := 2]
\]

take down a set of recursive equations for liveness analysis. Calculate its solution
using the work-list algorithm.

2. (CSE497 students only, 5 points; Exercise 2.4 of the book) Consider the following
program:

\[
[x := 1]; [x := x - 1]; [x := 2]
\]

Clearly \(x\) is dead at the exits from 2 to 3. But \(x\) is live at the exit of 1 even though
its only use is to calculate a new value for a variable that turns out to be dead. We
shall say that a variable is a faint variable if it is dead or if it is only used to calculate
new values for faint variables; otherwise it is strongly live. In the example \(x\) is faint
at the exits from 1, 2 and 3. Define a Data Flow Analysis that detects strongly live
variables. (Hint: For an assignment \([x := a]\) the definition of \(f_1(l)\) should be by cases
on whether \(x\) is in \(l\) or not.)