Exercise 5.8.

The following register-machine code is ambiguous, because the label `here` is defined more than once:

```
start
  (goto (label here))
here
  (assign a (const 3))
  (goto (label there))
here
  (assign a (const 4))
  (goto (label there))
there
```

With the simulator as written, what will the contents of register `a` be when control reaches `there`? Modify the `extract-labels` procedure so that the assembler will signal an error if the same label name is used to indicate two different locations.

Answer.

The contents of register `a` will be 3 when control reaches `there`. Observe that `extract-labels` accumulates to construct the instruction list and label table from the controller text in a bottom-up way. As the simulator processes this register-machine code, it places the first entry with label `here` preceding to the second one with the same label in the table. So the control of this machine will branch to the first entry with label `here` when encountered the `(goto (label here))` instruction, and directly branch to `there` after assigning the constant 3 to register `a`. The instruction sequence with the second `here` label is bypassed in this process.

We can augment `extract-labels` to check for duplication before adding an entry to the `labels` table, so that the assembler signal an error if the same label name is used to indicate two different locations.

```
(define (extract-labels text receive)
  (if (null? text)
      (receive '() '())
      (extract-labels (cdr text)
       (lambda (insts labels)
         (let ((next-inst (car text)))
           (if (symbol? next-inst)
               (let ((val (assoc next-inst labels)))
                 (if (not (null? val))
                     (error "Multiply defined label" next-inst)
                     (receive insts
                              (cons (make-label-entry next-inst insts)
                                    labels))))
               (receive (cons (make-instruction next-inst)
                              insts)
                        labels)))))))
```


Email address: informlarry@gmail.com